ALEXANDER ROBERT KUTZKE

ASSINALAMENTOS DE TESTES PARA UM ALGORITMO DE DIAGNÓSTICO EM NÍVEL DE SISTEMA PARA REDES DE SENSORES SEM FIO: UMA COMPARAÇÃO DE ABORDAGENS

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.
Orientadora: Profa. Dra. Andréa Weber

CURITIBA

2011

ALEXANDER ROBERT KUTZKE

ASSINALAMENTOS DE TESTES PARA UM ALGORITMO DE DIAGNÓSTICO EM NÍVEL DE SISTEMA PARA REDES DE SENSORES SEM FIO: UMA COMPARAÇÃO DE ABORDAGENS

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.
Orientadora: Profa. Dra. Andréa Weber

CURITIBA

2011

Ministério da Educação
Universidade Federal do Paraná
Programa de Pós-Graduação em Informática

PARECER

Nós, abaixo assinados, membros da Banca Examinadora da defesa de Dissertação de Mestrado em Informática, do aluno Alexander Robert Kutzke, avaliamos o trabalho intitulado, *"ASSINALAMENTOS DE TESTES PARA UM ALGORITMO DE DIAGNÓSTICO EM NÍVEL DE SISTEMA PARA REDES DE SENSORES SEM FIO: UMA COMPARAÇÃO DE ABORDAGENS"*, cuja defesa foi realizada no dia 15 de abril de 2011, às 13:30 horas, na Sala de Video-Conferência do Departamento de Informática do Setor de Ciências Exatas da Universidade Federal do Paraná. Após a avaliação, decidimos pela aprovação do candidato.

Curitiba, 15 de abril de 2011.

Profª. Dra. Andréa Weber
DINF/UFPR – Orientadora

Profª. Dra. Keiko Verônica Ono Fonseca
UTFPR – Membro Externo

Prof. Dr. Elias Procópio Duarte Jr.
DINF/UFPR – Membro Interno

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ
Setor de Ciências Exatas
Coordenação de Pós-Graduação em Informática

AGRADECIMENTOS

Aos meus pais e irmãos pelo seu carinho e apoio intermináveis durante toda minha vida e por terem proporcionado e incentivado meus estudos.

À minha querida Ana Cristina que me mostra a cada dia como a vida pode ser harmoniosa e intensa de felicidade e amor, e por sua ajuda e paciência, sempre presentes em todos os momentos.

A todos meus familiares, diretos e indiretos, por todo o carinho e pelo incentivo durante este trabalho.

À minha orientadora Andréa Weber por ter me guiado com muita atenção durante todo este trabalho, por ter me mostrado os caminhos da pesquisa e por toda paciência ao me ajudar na escrita de textos científicos.

Ao professor Stefano Chessa por sua ajuda e por seus esclarecimentos durante a definição dos algoritmos.

Aos professores Keiko Verônica Ono Fonseca e Elias Procópio Duarte Júnior por terem aceitado fazer parte da banca desta dissertação e pelos seus valiosos comentários durante o exame de qualificação deste trabalho.

Aos meus amigos pelos incontáveis momentos de diversão e alegria e pelo apoio sempre presente.

A todos que acreditaram em mim e me apoiaram.

A todas as pequenas coisas da vida que me fazem tão feliz, e que me guiaram até aqui.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Este trabalho se propõe a comparar três abordagens de construção de assinalamentos de testes para um algoritmo de diagnóstico em nível de sistema. As abordagens apresentadas visam o problema da detecção de alarmes falsos (falsos positivos) em uma rede de sensores sem fio onde os sensores monitoram o ambiente com o objetivo de gerar alarmes sobre a ocorrência de determinados eventos. Considere uma rede de sensores onde um conjunto de t sensores próximos geograficamente enviam sinais de alarme a uma unidade central da rede, com maior capacidade de processamento, chamada *sink*, informando a detecção de determinado fenômeno. Para garantir que os alarmes gerados não são falsos, o *sink* solicita a execução de testes mútuos entre os sensores presentes na região que contém os nodos que reportaram os alarmes. O resultado dos testes é enviado ao *sink* que, então, utiliza um algoritmo de *diagnóstico em nível de sistema* para identificar os sensores falhos. O algoritmo de diagnóstico é bem sucedido na execução desta tarefa se os testes executados pelos sensores são suficientes para alcançar determinada *diagnosticabilidade* do sistema, a qual depende de propriedades topológicas da rede de sensores e de certas condições presentes na literatura para formar assinalamentos de teste t-diagnosticáveis. Este trabalho apresenta três estratégias de testes que asseguram que a diagnosticabilidade desejada para o sistema seja alcançada com um consumo minimizado de energia. Resultados experimentais avaliam o comportamento das estratégias e comparam o consumo de energia apresentado entre elas em redes com diferentes topologias e densidades, com diferentes valores de t e com variações na distância entre os sensores que geram alarmes.

ABSTRACT

This work compares three test assignment approaches for a system-level diagnosis algorithm. The approaches address the problem of detecting false alarms (false positives) in a wireless sensor network (WSN) where the sensors monitor the environment with the objective of raising alarms about the detection of a predetermined event. Consider a WSN where a set of t sensors in a geographic neighborhood send alarms to the sink, a central unit of the network, informing about the event. To assure that the raised alarms are not false, the sink determines a set of mutual tests among the sensors in the region that contains the alarms. The tests results are sent to the sink that, in turn, uses a system-level diagnosis algorithm to identify the faulty sensors. The diagnosis algorithm is successful in detecting all the faulty units if the tests are enough to reach a determined system *diagnosability*, that depends on a set of topological properties of the sensor network and some diagnosability conditions present in the literature to generate t-diagnosable test assignments. This work presents three testing strategies that assure that a desired diagnosability is reached with a minimized energy consumption. Experimental results evaluate the behavior of the approaches and compare the energy consumption between them in different network topologies and densities, with different values of t and with variations in the distance between the sensors that raise alarms.

CAPÍTULO 1

INTRODUÇÃO

Diagnóstico em nível de sistema é uma área específica de tolerância a falhas, que se ocupa de determinar às unidades de um sistema o estado de todas as unidades como falhas ou sem-falha [7, 8]. Com aplicação em redes locais e de longa distância, entre outras, diagnóstico em nível de sistema apresenta também aplicação em um tipo específico de redes sem fio *ad hoc*, as redes de sensores [2, 9]. Redes de sensores são conjuntos densos de sensores de baixo custo, com recursos escassos de processamento, memória e bateria e que se comunicam por ondas de rádio com o objetivo de monitorar determinado fenômeno [2, 10].

Ao detectar algum comportamento pré-definido, como alterações no ambiente monitorado, sensores da rede são capazes de enviar mensagens de alarme às unidades centrais, com maior capacidade de processamento, comunicando o evento e seus dados. Em algumas aplicações, a geração de um alarme falso (falso positivo), ou seja, a informação errônea da ocorrência de um evento, pode ocasionar problemas e altos custos de recuperação. Um sistema capaz de detectar falhas em sensores e que possibilite a verificação da veracidade de um alarme gerado é necessário.

Este trabalho considera uma *rede de sensores sem fio* onde um conjunto de t sensores, geograficamente próximos, envia mensagens de alarme para uma unidade central. Com o objetivo de garantir que as mensagens de alarme não são falsas, a unidade central solicita uma série de testes entre os sensores presentes na região onde o alarme foi gerado. Resultados dos testes são coletados pela unidade central e analisados através de um algoritmo de *diagnóstico em nível de sistema* com o objetivo de detectar sensores falhos. Além disso, a estratégia de testes deve ser tal que miniminize o uso da energia pelos sensores, de forma a prolongar sua vida útil.

O restante deste Capítulo descreve brevemente o trabalho proposto. Na Seção 1.1

o conceito de redes de sensores sem fio é introduzido. A Seção 1.2 apresenta a área de diagnóstico em nível de sistema. Na Seção 1.3 a proposta deste trabalho é descrita. Por fim, a Seção 1.4 traz a organização do restante do texto.

1.1 Redes de Sensores Sem Fio

Redes de sensores sem fio ou WSN (*Wireless Sensor Networks*) são redes formadas por pequenos sensores conectados através de um canal de comunicação sem fio. Os sensores, ou nodos, que formam a rede são dispositivos de baixo custo, capazes de coletar e disseminar informações do ambiente onde estão inseridos [10]. Os sensores possuem capacidade local de processamento e, quando em grande número, podem disponibilizar uma visão geral de suas medições. Assim, uma análise mais detalhada do evento ou ambiente estudado é possível [1, 10].

O canal de comunicação sem fio permite aos nodos se organizarem, de forma autônoma e distribuída, em uma rede do tipo *ad hoc* [11]. Tal rede possibilita a troca de informação entre os nodos e também com uma ou mais unidades centrais que interagem com um usuário remoto [1].

Por terem capacidade de auto-organização, redes de sensores não necessitam de estratégias de posicionamento individual para cada nodo, permitindo assim o uso de um grande número de sensores na rede e também o monitoramento de áreas de difícil acesso ou ambientes inóspitos [2].

As redes de sensores sem fio possuem um grande conjunto de aplicações em diversos campos como: medicina, agricultura, militar, ambiental, engenharia entre outros [1]. Apesar de possibilitar uma grande gama de aplicações, as características das redes de sensores (hardware de baixo custo, pequena capacidade de processamento, de memória e de bateria, entre outros) também geram algumas dificuldades diferenciadas, para as quais soluções utilizadas em redes sem fio *ad hoc* não são suficientes ou satisfatórias [2]. Questões como economia de energia, localidade geográfica e roteamento de mensagens necessitam de estratégias especiais para que sejam utilizadas de forma eficiente em redes de sensores sem fio.

1.2 Diagnóstico em Nível de Sistema

Sistemas tolerantes a falhas necessitam de uma estratégia para identificar componentes falhos. A área de *Diagnóstico em Nível de Sistema* tem como objetivo estabelecer estratégias capazes de identificar unidades de um sistema como falhas ou sem-falha.

O primeiro modelo de diagnóstico em nível de sistema foi proposto por Preparata, Metze e Chien em 1967 [8]. Este modelo, conhecido como PMC, considera um sistema que consiste de unidades indivisíveis para o propósito de diagnóstico e capazes de realizar testes umas sobre as outras. Tais unidades devem ter também capacidade de reportar com precisão o estado de outras unidades do sistema. No modelo PMC, testes realizados por unidades sem-falha são confiáveis enquanto testes realizados por unidades falhas têm resultado arbitrário.

O sistema, no modelo PMC, é representado por um grafo completo. O grafo de testes é um grafo direcionado onde os vértices são as unidades do sistema, e uma aresta partindo do vértice u_i para o vértice u_j indica que u_i realiza testes sobre u_j. O conjunto dos resultados de todos os testes executados é chamado de *síndrome* e é analisado por uma unidade central que realiza o diagnóstico.

Ainda, no modelo PMC um sistema S é dito *t-diagnosticável* se com até t unidades falhas presentes em S, a unidade central é capaz de realizar o diagnóstico. Diz-se que a *diagnosticabilidade* de um sistema S é igual a t se S for t-diagnosticável.

O trabalho de Preparata, Metze e Chien demonstra duas condições necessárias para que um sistema seja t-diagnosticável: $(c1)$ o número N de nodos do sistema deve satisfazer $N \geq 2 * t + 1$, onde t é a diagnosticabilidade do sistema; e $(c2)$ cada unidade deve ser testada por pelo menos t outras unidades.

Posteriormente, Hakimi e Amin demonstraram que as condições acima são necessárias e suficientes para que um sistema seja t-diagnosticável se não houverem testes mútuos entre as unidades do sistema [12]. Para casos onde existem testes mútuos, uma terceira condição é demonstrada, para a qual um corolário é dado: seja G um grafo direcionado que representa o sistema S, e $k(G)$ a conectividade do grafo G; se $k(G) \geq t$, então S é t-diagnosticável.

Embora outras abordagens e algoritmos de diagnóstico tenham sido propostos posteriormente [13, 14, 15, 16, 17, 18, 19], o fato do modelo PMC depender de uma unidade central para fazer o diagnóstico encontra aplicação em redes de sensores sem fio devido à presença do *sink*, que é uma unidade central capaz de coletar e processar informações, entre elas, as de diagnóstico.

1.3 Proposta deste Trabalho

Este trabalho apresenta uma comparação entre abordagens de testes para o problema da detecção de alarmes falsos (falsos positivos) em uma rede de sensores sem fio onde os sensores monitoram o ambiente com o objetivo de gerar alarmes sobre a ocorrência de determinados eventos. É adotado um modelo de diagnóstico em nível de sistema, onde os sensores existentes na região em que o alarme foi gerado cooperam para a execução de testes entre si. O resultado obtido através dos testes é utilizado como entrada para um algoritmo de diagnóstico que identifica os sensores falhos e, assim, confirma ou rejeita o alarme reportado.

Mais especificamente, dada uma região da rede de sensores onde t mensagens de alarme foram geradas, este trabalho propõe formas de organização dos testes executados entre os nodos, as quais tem como objetivo possibilitar que o grafo que representa o conjunto de testes seja t-diagnosticável. Isso envolve, entre outras questões, formas de delimitar a região onde os alarmes foram gerados, bem como a organização da rede de forma a evitar testes recíprocos.

As estratégias apresentadas neste trabalho consideram o gasto energético global utilizado pela rede e o gasto energético de cada sensor na execução dos testes. Os grafos de testes gerados pelas estratégias buscam minimizar o consumo de energia, para que a vida útil da rede seja maximizada. Dentre as estratégias, uma é distribuída, no sentido em que os próprios sensores definem o grafo de testes. Nas demais, o *sink* analisa o posicionamento dos sensores e gera o grafo de testes, o qual é comunicado à rede.

Nas estratégias propostas neste trabalho, a unidade central presente na rede de sensores, ou *sink*, é responsável, também, pela tarefa de decodificação da síndrome. Desta forma

cada nodo deve repassar suas informações de testes ao *sink*. Para que essas informações alcancem a unidade central, é necessária uma estratégia de roteamento de mensagens que permita que as rotas não possuam nodos diagnosticados como falhos. A abordagem apresentada por Karp e Kung [20] pode ser utilizada.

Por fim, a decodificação da síndrome no *sink* pode ser realizada pelo algoritmo apresentado por Dahbura e Masson [21]. Outra alternativa pode ser um diagnóstico aproximado obtido através do algoritmo de Caruso et al [22]. O *sink* pode, ainda, utilizar métodos de correlações de medidas para realizar a decodificação.

Simulações que comparam o comportamento e o consumo de energia de cada abordagem são apresentadas neste trabalho. Os experimentos realizados contemplam diferentes configurações de redes e cenários de geração de alarmes.

1.4 Organização deste Trabalho

O presente trabalho está organizado como segue: o Capítulo 2 trata de redes de sensores sem fio. No Capítulo 3 a área de diagnóstico em nível de sistema é apresentada. O Capítulo 4 descreve as estratégias de testes propostas. Os Capítulos 5 e 6 trazem a descrição dos experimentos executados e a conclusão do trabalho, respectivamente.

CAPÍTULO 2

REDES DE SENSORES SEM FIO

Um sensor, segundo uma definição simples, é um dispositivo capaz de observar e registrar algum tipo de fenômeno. Sensores são utilizados nas mais diferentes áreas como militar, medicina, engenharia, meteorologia, segurança, entre outras [2].

A constante evolução tecnológica de dispositivos eletrônicos tem possibilitado, cada vez mais, a criação de pequenos sensores de baixo custo, capazes de executar processamento local de informações e de realizar comunicação sem fio. A união de sensores com estas características, através de algum protocolo de comunicação, e sua ação coordenada, conduzem ao conceito de *redes de sensores sem fio* [10].

Este Capítulo disserta sobre redes de sensores e está organizado da seguinte maneira: a Seção 2.1 apresenta a definição de redes de sensores sem fio e uma comparação com outros métodos de sensoriamento existentes. A Seção 2.2 mostra várias aplicações existentes para redes de sensores. Uma descrição de arquitetura de sensores é apresentada na Seção 2.3. Na Seção 2.4 as características e desafios presentes no estudo de redes de sensores são descritas. Por fim, a Seção 2.5 traz detalhes sobre arquiteturas de comunicação utilizadas em redes de sensores.

2.1 Definição

Redes de sensores sem fio ou WSN (*Wireless Sensor Networks*) são redes densas formadas por pequenos sensores de baixo custo. Os sensores, ou nodos, presentes na rede trabalham de forma coordenada e colaborativa para obter um sensoriamento ideal do fenômeno. Para trabalhar desta forma, sensores possuem dispositivos que permitem comunicação sem fio e processamento local de informações. As informações coletadas pelos nodos da rede são transmitidas para uma ou mais unidades centrais, ou *sinks*, que servem como ponto de acesso à rede de sensores para usuários remotos [2, 10, 1]. A Figura 2.1 mostra um

exemplo de organização de uma rede de sensores.

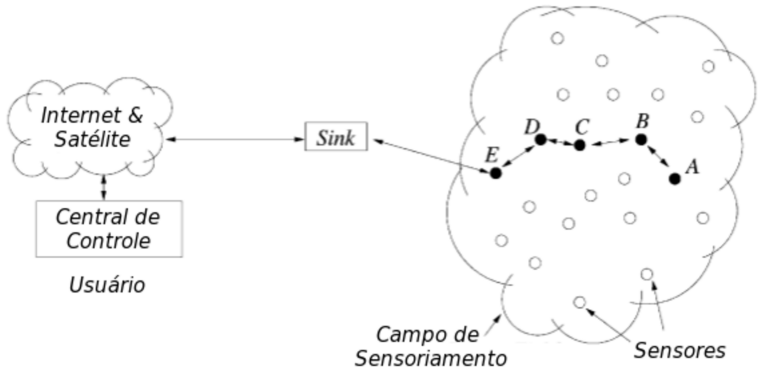


Figura 2.1: Exemplo de organização de redes de sensores [1].

Sensores podem atuar no monitoramento de algum fenômeno. Para tal tarefa, os nodos da rede, ao detectarem algum comportamento pré-estabelecido do fenômeno em observação, enviam mensagens de alarme ao *sink*, que então, informa ao usuário remoto sobre a ocorrência do evento.

2.1.1 Sistemas Tradicionais de Sensoriamento

Redes de sensores sem fio representam um grande avanço sobre a tecnologia tradicional de sensores, a qual é composta por duas estratégias principais [2]: (1) através de grandes equipamentos, com capacidade de sensoriamento e de processamento, posicionados o mais próximo possível do evento ou ambiente a ser analisado; ou (2) com unidades menores capazes de executar apenas a tarefa de sensoriamento, depositadas em posições geográficas cuidadosamente definidas. Estas unidades fornecem uma série temporal de medições, que são, então, processadas por grandes nodos centrais.

Na primeira abordagem, por serem compostos por grandes equipamentos, posicionar os dispositivos sensoriais próximos do fenômeno se torna uma tarefa nem sempre possível, diminuindo assim a qualidade dos dados registrados pelo sensor. Isso acontece pelo fato de que, além do fenômeno, outras características do ambiente também podem ser captadas, gerando ruído nos dados coletados. Desta forma os equipamentos utilizados devem analisar o sinal coletado e realizar operações sobre o mesmo para que apenas dados do

fenômeno desejado sejam considerados. Porém esta tarefa é cara e complexa, tornado esta abordagem de sensoriamento não funcional para alguns casos.

Na segunda abordagem, os sensores podem ser posicionados próximos do fenômeno, garantindo assim uma melhor qualidade dos dados obtidos. Porém por conterem apenas a capacidade de realizar sensoriamento, tais sensores devem ser resgatados após o registro do fenômeno para que os dados sejam analisados por unidades centrais de processamento. Desta forma, o posicionamento de cada sensor deve ser conhecido, e claro, alcançável para que os dispositivos possam ser recuperados. Após recolher todos os sensores, a análise e a fusão dos dados é realizada, gerando o resultado final do sensoriamento.

Redes de sensores, em contrapartida, são formadas por um grande número de nodos depositados o mais próximo possível do fenômeno, sem que seja necessário conhecer ou pré-determinar o posicionamento individual de cada sensor. Isso permite que um posicionamento aleatório dos sensores seja realizado sobre o fenômeno, porém cria a necessidade de que a rede e seus dispositivos tenham capacidade de auto-organização. Esta característica é alcançada pela capacidade de comunicação e de processamento de cada sensor.

2.2 Aplicações

As características existentes nas redes de sensores sem fio permitem uma grande aplicabilidade em diversos campos como: medicina, ambiental, militar, residencial, engenharia entre outros [1]. São apresentados a seguir alguns exemplos de diferentes aplicações das redes de sensores nesses campos.

Saúde

Trabalhos como [23, 24] mostram inúmeras aplicações para redes de sensores no campo de saúde e bem-estar. Sensores podem ser utilizados para realizar medições remotas e discretas de parâmetros fisiológicos de um paciente, e enviar dados ao hospital se alguma alteração for percebida.

Outras possíveis aplicações seriam o controle da administração de remédios e o rastre-

amento de médicos e pacientes em hospitais.

Ambiental

Sensores podem ser utilizados no monitoramento e controle de diferentes fatores na área ambiental. Controle e detecção de incêndios florestais, possíveis inundações, monitoramento da biodiversidade de alguma localidade ou ainda auxílio no controle de grandes plantios são apenas alguns exemplos de aplicações.

Redes de sensores podem ser depositadas em grandes áreas florestais para que inícios de incêndio possam ser notificados para usuários remotos. Desta forma, grandes incêndios podem ser evitados e a localização exata do início deles pode ser facilmente conhecida, favorecendo muito o trabalho de contenção do fogo. A mesma estratégia se aplica para inundações, onde sensores podem monitorar condições climáticas e informar o nível da água em diferentes regiões, podendo, assim, prever uma inundação [2].

Sensores também podem ser usados para registrar e rastrear espécies presentes em regiões de difícil acesso, permitindo um melhor conhecimento da biodiversidade destas regiões [25].

Na agricultura, redes de sensores tem como aplicabilidade o monitoramento de grandes áreas de plantio, possibilitando a detecção das necessidades e características específicas de diferentes pontos da plantação.

Militar

A área militar é um grande campo de aplicação. Questões como monitoramento do campo de batalha à procura de tropas inimigas ou alguma atividade suspeita, ou ainda a detecção de ataque com armas químicas, são apenas alguns exemplos das muitas aplicações das redes de sensores na área militar. A natureza distribuída, o baixo custo dos sensores e a tolerância a falhas presentes nas redes de sensores são características que favorecem o uso desta tecnologia em operações militares [2, 1].

Residencial

Redes de sensores podem automatizar ambientes residenciais e torná-los, de certa forma, inteligentes. Dispositivos domésticos comuns como aparelhos de micro-ondas, aspiradores de pó, geladeiras, televisores e outros, podem ser equipados com sensores. Tais sensores formam uma rede doméstica tornando possível a comunicação entre os dispositivos e até mesmo permitindo ao usuário um controle à distância dos aparelhos da casa.

Outra possível abordagem é a de tornar cada cômodo da casa uma pequena rede, com sensores espalhados pelos dispositivos presentes. Estas redes se comunicam entre si, permitindo um alto nível de automatização das tarefas domésticas.

Sensores ainda podem ser usados para monitorar uma série de detalhes como temperatura, umidade, quantidade de pó, presença de pessoas, entre outros, dentro da casa.

Engenharia

No campo da engenharia civil, sensores podem ser utilizados para monitorar o estado de pontes, prédios e outros tipos de construções podendo antecipar possíveis problemas estruturais.

2.3 Arquitetura de um Sensor

Nodos utilizados em redes de sensores são divididos internamente em quatro unidades principais: unidade de processamento, unidade de comunicação, unidade de sensoriamento e unidade de energia [10]; um sensor pode, dependendo da aplicação, ser composto por mais unidades, como, por exemplo, um sistema de localização. A Figura 2.2 apresenta um exemplo de arquitetura de sensores.

A *unidade de processamento* consiste de um microprocessador (ou unidade microcontroladora) responsável por qualquer operação computacional executada pelo sensor. Operações como execução de protocolos de comunicação, análise de dados obtidos pela unidade de sensoriamento e organização da rede em cooperação com os demais nodos são exemplos de tarefas em que a unidade de processamento é utilizada. Mais especificamente,

a existência de um microprocessador em cada sensor permite que a rede possa trabalhar de forma eficiente e autônoma.

O uso do microprocessador em cada nodo é controlado de forma a economizar energia. Além de ser projetado para tarefas de baixa complexidade, diferentes modos de operação para a unidade de processamento podem ser criados para que o consumo de energia seja o menor possível.

A *unidade de comunicação* é composta por dispositivos capazes de transmitir e receber mensagens. Um exemplo de dispositivo de comunicação em sensores são as transmissões via rádio de pequeno alcance. Esta unidade tem a função de realizar a comunicação de cada nodo com seus sensores vizinhos e com as unidades centrais da rede. As operações executadas pela unidade de comunicação consomem uma grande quantidade de energia, pois necessitam garantir boa qualidade de transmissão e alcance para que a rede trabalhe de forma satisfatória.

A tarefa de coletar e registrar dados do fenômeno estudado é executada pela *unidade de sensoriamento*. Esta unidade contem um ou mais tipos de sensores conforme a natureza do sensoriamento a ser executado.

A *unidade de energia* de um sensor consiste, de forma geral, de uma bateria, ou seja, cada sensor possui uma quantidade limitada de energia. Desta forma, redes de sensores têm como uma das principais restrições a economia de energia.

Em [1] é apresentada uma comparação entre diferentes arquiteturas de sensores existentes. Em algumas arquiteturas, o raio de alcance da unidade de comunicação dos sensores pode ser de até 300 metros de distância. Fato que garante aplicações para redes de sensores nos mais diversos ambientes.

2.4 Características e Restrições das Redes de Sensores

Diferentemente de redes sem fio WLAN (*Wireless Local Area Network*), redes sem fio *ad hoc* [11] não possuem uma unidade responsável pela coordenação da comunicação da rede. Em redes sem fio *ad hoc* a comunicação e organização do nodos é realizada através da cooperação entre os próprios nodos.

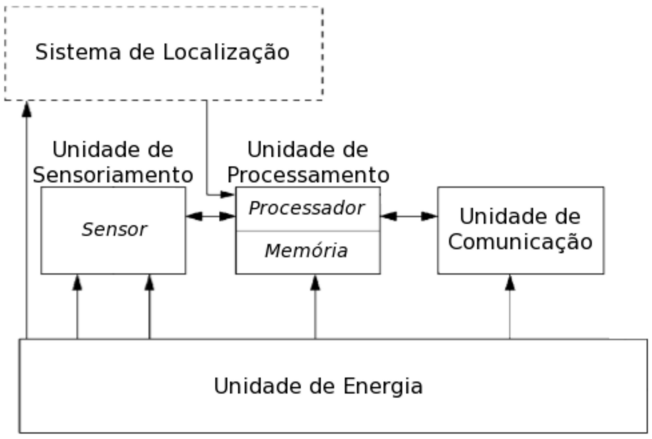


Figura 2.2: Exemplo de arquitetura de sensores [2].

Apesar de possuir muitas características oriundas de redes sem fio *ad hoc*, muitos dos protocolos e algoritmos propostos para estas redes não atendem as características específicas presentes nas redes de sensores. Comparativamente a redes sem fio *ad hoc* [2]:

- redes de sensores apresentam número elevado de nodos;
- redes de sensores apresentam alta densidade de nodos;
- sensores têm capacidade limitada de processamento, energia e memória;
- sensores são sujeitos a falhas;
- a topologia de uma rede de sensores pode mudar frequentemente;
- a maioria dos sensores utiliza o paradigma de comunicação de *broadcast* de mensagens, enquanto grande parte das redes *ad hoc* são baseadas na comunicação *ponto-a-ponto*;
- sensores podem não possuir identificação individual, devido principalmente ao grande número de nodos e ao *overhead* causado.

São apresentadas a seguir algumas características e desafios presentes na área de redes de sensores.

2.4.1 Tolerância a Falhas

Sensores são dispositivos sujeitos a falhas. Por essa razão redes de sensores devem ser capazes de trabalhar corretamente mesmo com a possibilidade de que alguns de seus nodos venham a falhar. Algumas causas de falhas em sensores podem ser: falta de energia, dano físico, interferência, entre outras [2].

As características presentes no ambiente onde a rede deve funcionar implicam no grau de tolerância a falhas que o sistema deve manter. Por exemplo, em um ambiente doméstico, uma rede que monitora a temperatura de diferentes regiões da casa não necessita de grandes preocupações com alguns tipos falhas, pois o ambiente não oferece muitas perturbações. Porém em outros usos, como em um campo de batalha, as redes de sensores devem ter algoritmos e protocolos desenvolvidos para trabalharem mesmo sob uma grande taxa de falha de seus nodos.

2.4.2 Escalabilidade

Uma rede de sensores pode ser composta por centenas, milhares ou até milhões de nodos [1]. Logo, protocolos, algoritmos e a própria rede devem ser capazes de operar de forma satisfatória mesmo com este alto número de sensores presentes na rede.

Além disso, protocolos devem ser capazes de utilizar a grande densidade da rede para o melhor gerenciamento dos recursos presentes em cada nodo. Por exemplo, uma alternativa de economia de energia que uma rede com grande densidade de nodos pode propiciar é a possibilidade de que alguns nodos sejam desligados por certos períodos de tempo enquanto outros realizam suas tarefas normalmente.

2.4.3 Custos de Produção

Um grande desafio para o uso de redes de sensores ainda é o custo de produção dos sensores. Devido ao grande número de sensores utilizados nas redes e à grande probabilidade de perda ou de danos causados sobre eles, o custo de produção deve ser extremamente pequeno.

Apenas para comparação, dispositivos de comunicação com a tecnologia *Bluetooth* podem custar cerca de US$10, porém estudos mostram que o preço ideal para um nodo de uma rede de sensores seria algo muito menor do que US$1 [1], pois tal preço permitiria uma produção em larga escala e facilitaria aplicações. Ou seja, mesmo dispositivos *Bluetooth* que são considerados equipamentos baratos custam quase 10 vezes mais do que o preço considerado ideal para um sensor.

2.4.4 Ambiente de Operação

Redes de sensores devem ser capazes de operar em diferentes ambientes. Desde os mais simples e favoráveis como uma residência, até os mais complicados e inóspitos como oceanos, tornados, campos contaminados biológica ou quimicamente, campos de batalha entre outros.

Para cada situação apresentada diferentes tipos de sensores capazes de suportar as condições do ambiente onde estão inseridos serão utilizados. Porém as redes devem ser capazes de operar em qualquer um desses ambientes.

2.4.5 Limitações de Hardware

É interessante para as redes de sensores que os nodos tenham um tamanho reduzido, pois o número de nodos presentes na rede é muito elevado. Conforme apresentado na Seção 2.3, a arquitetura de um sensor pode ser dividida em quatro partes: unidade de processamento, unidade de comunicação, unidade de sensoriamento e unidade de energia. Essas quatro partes devem ser pequenas o suficiente para que o sensor não exceda o tamanho desejado.

A maioria dos algoritmos de roteamento em redes de sensores assumem que cada nodo sabe sua própria posição geográfica. Para que isso ocorra, a forma mais utilizada é disponibilizar um receptor GPS em cada nodo. Porém receptores GPS causam um maior consumo de energia e um aumento no tamanho dos sensores, ambos indesejados. Uma abordagem alternativa consiste em apenas alguns sensores possuírem um receptor GPS e este nodo auxiliar seus vizinhos a definir suas posições geográficas relativas a ele.

Em [26] é apresentada uma visão geral de várias técnicas de localização sem o uso de GPS. Entre elas:

Temporização: A posição de um sensor pode ser estimada baseada na distância dele para um ponto de referência. Tal distância é calculada com base no tempo necessário para realizar a comunicação entre os dois pontos.

Potência do sinal: À medida que um sinal é propagado sua potência diminui proporcionalmente à distância percorrida, o que permite o cálculo da distância entre dois dispositivos.

Direcionalidade: Os ângulos de cada ponto de referência em relação a um nodo podem determinar sua localização.

Localização baseada por proximidade: Um nodo pode solicitar as posições de cada um de seus vizinhos, e a partir do cálculo do centroide destas posições pode definir uma aproximação de sua localização.

2.4.6 Meio de Comunicação

A escolha de um meio de comunicação sem fio utilizado pelos nodos em uma rede de sensores é uma questão de grande importância. Alguns possíveis meios podem ser rádio, infra-vermelho ou óptico [2]. Porém para permitir um funcionamento padrão, a forma de comunicação deve estar disponível igualmente em todos os ambientes de utilização.

Transmissões via rádio necessitam de faixas de frequências disponíveis para operar. Uma opção para as redes de sensores é o uso das faixas ISM (*Industrial, Scientific, Medical*), que são disponíveis globalmente e livres de licença. Por outro lado, limitações de energia e interferências com outras aplicações existentes nas mesmas faixas de frequência ainda são problemas existentes.

Comunicação realizada através de infra-vermelho apresenta as vantagens de ser mais barata e possuir uma implementação mais simples, porém traz a necessidade de existir uma linha de visão entre os dispositivos que participam da conexão, fato que não ocorre com

frequência em redes de sensores. A comunicação através de dispositivos ópticos também é uma outra opção para redes de sensores, porém apresenta a mesma desvantagem em depender da existência de uma linha de visão entre os sensores da rede.

2.4.7 Consumo de Energia

Sensores são equipados com uma fonte limitada de energia. Em vários ambientes a troca ou reposição desta fonte é impossível. Por esse motivo o tempo de funcionamento que um sensor desempenha está diretamente ligado à maneira com que a energia de cada nodo é administrada.

Diferentemente das redes *ad hoc*, onde a economia de energia é desejável mas não é a consideração primária, redes de sensores visam primeiramente o baixo consumo de energia para que o tempo de vida da rede seja maximizado. Desta forma, o hardware presente nos sensores, os protocolos e os algoritmos utilizados são todos projetados com foco principal no consumo de energia.

O gasto de energia em um sensor ocorre com a execução de 3 diferentes tarefas [2]: sensoriamento, comunicação e processamento de dados. A tarefa de sensoriamento não representa um grande consumo de energia, porém este consumo é proporcional às complicações causadas pelo ambiente onde o nodo está inserido. Se o ambiente causa ruído ou atrapalha de alguma forma o sensoriamento, tarefas de filtragem ou melhora do sinal obtido podem ser requeridas, causando, assim, um maior gasto de energia. A tarefa mais custosa em termos da energia presente em um nodo é a tarefa de comunicação. A transmissão via rádio por exemplo, é a operação que mais consome energia.

Como as mensagens enviadas entre os nodos são pequenas e esporádicas na maioria dos casos de redes de sensores [2], uma alternativa para um menor consumo de energia é desligar a interface de comunicação enquanto não utilizada. Porém as tecnologias disponíveis hoje sofrem um consumo de energia significativo ao realizar o processo de inicialização da interface.

O processamento de dados executado nos sensores gasta uma quantidade menor de energia do que a tarefa de comunicação. Assim, protocolos e algoritmos visam executar o

máximo de processamento local nos sensores para que o número e tamanho das mensagens enviadas sejam minimizados.

2.4.8 Topologia e Roteamento

O grande número de sensores presentes na rede, a mobilidade e as limitações de hardware e de energia são detalhes que tornam o controle de topologia de uma rede de sensores uma tarefa complexa.

É necessário que exista o controle ou conhecimento da topologia da rede para que a maioria dos protocolos, especialmente de roteamento, funcionem corretamente.

Protocolos convencionais de roteamento possuem sérias complicações quando utilizados sob as limitações de energia existentes nas redes de sensores [10]. Estes protocolos utilizam com frequência a técnica de inundação de mensagens na rede, onde um nodo, ao receber uma mensagem, a repassa para todos os seus vizinhos. Porém esta técnica causa um grande consumo de energia e de memória, dificultando seu uso com sensores.

Em [27] é apresentada uma técnica que permite que o roteamento das mensagens seja realizado visando a minimização no gasto de energia causado pelo envio da mensagem. Ou seja, não é escolhido como rota necessariamente o menor caminho, mas sim o caminho que possivelmente causa o menor consumo de energia. O consumo de energia utilizado por uma dada mensagem é calculado com base em heurísticas pré-determinadas.

Ainda nesta técnica, as rotas são definidas a cada nova mensagem. Desta forma, a probabilidade de que uma mesma rota seja utilizada mais de uma vez é diminuída, possibilitando que, um consumo de energia mais homogêneo seja realizado.

Outros estudos como [28, 29, 30, 31] apresentam protocolos e técnicas de roteamento para redes de sensores.

2.5 Arquitetura de Comunicação

Esforços têm sido feitos para a criação de padrões que permitam uma organização comum para redes de sensores. Desta forma, aplicações podem ser desenvolvidas com uma maior

liberdade visto que as implementações não estarão ligadas a uma arquitetura específica de rede, mas sim a um padrão pré-especificado e comum a outras aplicações.

Em [2] é apresentada uma pilha de protocolos (Figura 2.3) sobre a qual as redes de sensores podem trabalhar. Esta pilha consiste das seguintes partes: camada de aplicação, camada de transporte, camada de rede, camada de enlace, camada física e planos de energia, de mobilidade e de tarefa.

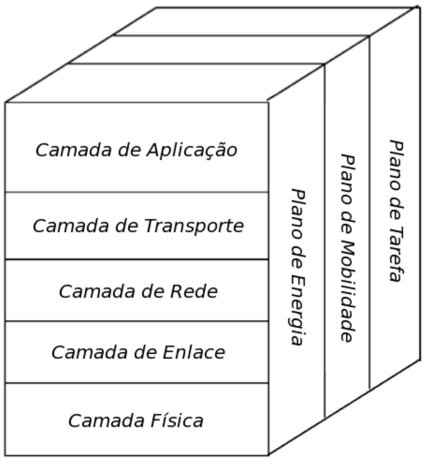


Figura 2.3: Pilha de protocolos para redes de sensores [2].

A camada de aplicação é responsável pelas diferentes aplicações que podem ser criadas para utilizar a rede de sensores. A camada de transporte coordena o fluxo de dados da rede criado pela aplicação. A camada de rede realiza o roteamento das mensagens geradas pela camada de transporte. O acesso ao meio de comunicação e a interação nodo a nodo são controlados pela camada de enlace. Por fim, a camada física é responsável pelas técnicas de envio e recebimento de mensagens e sinais em cada dispositivo da rede.

Os planos de energia, de mobilidade e de tarefa permitem aos nodos trabalharem de forma coordenada. Esses três planos monitoram a energia e a mobilidade dos sensores e distribuem diferentes tarefas entre si de modo a otimizar o sensoriamento e diminuir o consumo total de energia.

Wired Integrated Network Sensors (WINS) e *Sensor Information Networks Architecture* (SINA) são exemplos de arquiteturas criadas para redes de sensores. Tais arquiteturas podem ser mapeadas para a pilha de protocolos apresentada [2].

2.5.1 Padrões IEEE 802.15.4 e ZigBee

ZigBee Alliance é uma associação de companhias que trabalham juntas para o desenvolvimento de padrões (e produtos) com o objetivo de criar redes sem fio confiáveis, com um bom custo benefício e de baixo consumo de energia [32]. A intenção da organização ZigBee é que sua tecnologia esteja presente em um grande número dispositivos em breve [1].

O padrão ZigBee define as camadas mais altas da pilha de protocolos. A camada de aplicação disponibiliza um *framework* para desenvolvimento de aplicações distribuídas e para comunicação sobre as redes de sensores, enquanto que a camada de rede é responsável por fornecer e organizar o roteamento sobre uma rede de comunicação.

A camada de rede fornecida pelo padrão ZigBee foi desenvolvida para trabalhar sobre o padrão IEEE 802.15.4 e suas funcionalidades. O padrão 802.15.4 desenvolvido pelo IEEE tem como objetivo fornecer definições para o conjunto de características das camadas física e de enlace para redes do tipo LR-WPAN (*Low-Rate Wireless Personal Area Networks*). Redes LR-WPAN têm como características: facilidade de instalação, custo extremamente baixo, baixo consumo de energia e protocolos simples e flexíveis [33, 1, 34]. Juntos, os padrões ZigBee e 802.15.4 formam uma promissora arquitetura para uso em redes de sensores [35].

CAPÍTULO 3

DIAGNÓSTICO EM NÍVEL DE SISTEMA

Sistemas computacionais precisam ser confiáveis. Muitos ambientes exigem aplicações com altos níveis de confiabilidade. A propriedade *dependability* [7] de um sistema, traduzida como *confiança no funcionamento*, é definida como a capacidade do sistema de prover serviços em que o usuário pode confiar de forma justificável. *Serviço* é o comportamento do sistema como percebido pelos seus usuários.

A propriedade de *confiança no funcionamento* é composta por cinco atributos principais [7]: disponibilidade (*availability*), confiabilidade (*reliability*), segurança (*safety*), integridade (*integrity*), manutenibilidade (*maintainability*). Disponibilidade consiste na capacidade do sistema de oferecer serviços de forma correta; confiabilidade significa estar apto a fornecer serviço contínuo; segurança é propriedade do sistema de evitar consequências desastrosas; integridade é a capacidade do sistema de não sofrer alterações impróprias; e manutenibilidade se refere à capacidade do sistema de permitir modificações e reparos.

Em [7] são definidos, ainda, os termos falha (*fault*), erro (*error*) e defeito (*failure*). Defeito é a produção de uma resposta incorreta pelo sistema para uma dada entrada; erro, por sua vez, é algo no sistema que pode ou não promover o defeito; e falha é a causa provável ou hipotética do erro.

Técnicas ou meios utilizados para atingir *confiança no funcionamento* podem ser agrupados em quatro categorias diferentes: prevenção de falhas (*fault prevention*), tolerância a falhas (*fault tolerance*), remoção de falhas (*fault removal*) e previsão de falhas (*fault forecasting*).

Tolerância a falhas, especificamente, é dividida em duas áreas principais: detecção de erros (*error detection*), responsável pela identificação da presença de erros, e recuperação de erros (*error recovery*) que visa a transição de um estado do sistema com erros e possíveis

falhas, para um estado sem erros. A área de recuperação de erros é composta pelas técnicas de tratamento de erros (*error handling*), que tem como função eliminar os erros presentes no estado do sistema, e tratamento de falhas (*fault handling*) que previne que falhas que já ocorreram aconteçam novamente. O tratamento de falhas por sua vez, possui, entre outras, a área de diagnóstico (*diagnosis*), que é responsável por identificar falhas em termos de localização e natureza.

Sistemas tolerantes a falhas, portanto, necessitam de estratégias para identificar componentes falhos. A abordagem de diagnóstico em nível de sistema tem como objetivo identificar o estado de todas as unidades do sistema como falhas ou sem-falha.

Neste Capítulo são abordados conceitos e algoritmos de diagnóstico em nível de sistema. A Seção 3.1 descreve modelos e algoritmos de diagnóstico em nível de sistema para redes completamente conectadas. Na Seção 3.2 são apresentados detalhes da área de diagnóstico baseado em comparações. As tendências atuais em diagnóstico em nível de sistema são abordadas na Seção 3.3. Por fim, a Seção 3.4 disserta sobre algoritmos de diagnóstico para redes de sensores.

3.1 Diagnóstico de Redes Completamente Conectadas

As Seções a seguir apresentam diferentes abordagens de diagnóstico em nível de sistema para redes completamente conectadas, ou seja, para redes representáveis por grafos completos.

3.1.1 Modelo PMC

O primeiro modelo de diagnóstico em nível de sistema foi proposto por Preparata, Metze e Chien em 1967 [8]. Este modelo, conhecido como PMC, assume um sistema S que consiste de um conjunto de N unidades independentes, u_0, u_1, ..., u_{N-1}. No presente trabalho, as unidades do sistema S também são referenciadas como *nodos*, e uma unidade u_i como *nodo i*.

No modelo PMC, as unidades presentes no sistema S são capazes de realizar testes

umas sobre as outras e reportar resultados de testes. Testes realizados por unidades sem-falha são confiáveis enquanto testes realizados por unidades falhas têm resultado arbitrário. No modelo PMC os testes funcionam com base em um conjunto de estímulos sobre a unidade testada, cujas respostas permitem ao nodo testador classificar a unidade testada como falha ou sem-falha.

O sistema, no modelo PMC, é representado por um grafo completo. Os testes executados entre as unidades do sistema podem ser representados na forma de um grafo direcionado. Neste grafo, conhecido como grafo de testes, os vértices são as unidades do sistema, e uma aresta partindo do vértices u_i para o vértice u_j indica que u_i realiza testes sobre u_j. O conjunto dos resultados de todos os testes executados é chamado de *síndrome*. Cada aresta presente no grafo possui um peso, o valor binário $a_{i,j}$, que representa o resultado obtido pelo teste realizado. Se uma unidade i, sem-falha, testa a unidade j como falha, então $a_{i,j} = 1$; por outro lado se a unidade i testa a unidade j como sem-falha, então $a_{i,j} = 0$.

O diagnóstico do sistema S depende da organização dos testes e também do número de unidades falhas existentes. No modelo PMC, um sistema S é dito t-diagnosticável, ou t-diagnosticável em um passo, se com até t unidades falhas presentes em S, a unidade central é capaz de realizar o diagnóstico. Diz-se que a diagnosticabilidade de um sistema S é igual a t se S for t-diagnosticável. Preparata et al. ainda definem sistemas sequencialmente t-diagnosticáveis como sistemas em que pelo menos uma unidade falha pode ser identificada e recuperada, ou seja, passada do estado falho para sem-falha, ou substituída. Assim, o testes continuam até que todas as unidades falhas sejam diagnosticadas de forma sequencial.

O trabalho de Preparata, Metze e Chien demonstra duas condições necessárias para que um sistema seja t-diagnosticável: $(c1)$ o número N de nodos do sistema deve satisfazer $N \geq 2 * t + 1$, onde t é a diagnosticabilidade do sistema; e $(c2)$ cada unidade deve ser testada por pelo menos t outras unidades.

Posteriormente, Hakimi e Amin demonstraram que as condições acima são necessárias e suficientes para que um sistema seja t-diagnosticável se não houverem testes mútuos

entre as unidades do sistema [12]. Para casos onde existem testes mútuos, uma terceira condição deve ser satisfeita, para a qual um corolário é dado: seja G um grafo direcionado que representa o sistema S, e $k(G)$ a conectividade do grafo G; se $k(G) \geq t$, então S é t-diagnosticável.

Preparata et al. assumem que no modelo PMC unidades sem-falha sempre executam testes corretos, e que resultados obtidos através de testes realizados por unidades falhas tem resposta arbitrária. Outro modelo de diagnóstico, proposto por Barsi, Grandoni e Maestrini em [36], emprega o mesmo grafo de testes do modelo PMC, porém assume que uma unidade falha nunca avalia uma outra unidade falha como sem-falha. Esta asserção é conhecida como *invalidação assimétrica* de testes, enquanto que a asserção usada no modelo PMC é chamada de *invalidação simétrica* de testes.

3.1.2 Problemas Básicos do Diagnóstico em Nível de Sistema

A área de diagnóstico em nível de sistema pode ser dividida em três problemas básicos [37]: problema da caracterização (*characterization problem*), problema da diagnosticabilidade (*diagnosability problem*) e problema do diagnóstico (*diagnosis problem*).

O problema da caracterização consiste em encontrar condições necessárias e suficientes para alcançar a diagnosticabilidade desejada ao sistema. Já o problema da diagnosticabilidade consiste em, dado um conjunto de testes para o sistema, determinar se o sistema é t-diagnosticável. Por fim, o problema do diagnóstico é definido como a determinação das unidades falhas presentes no sistema a partir de sua síndrome.

3.1.3 Diagnóstico Adaptativo

No diagnóstico introduzido pelo modelo PMC o conjunto de testes a ser executado por cada nodo é previamente definido e permanece inalterado até o final do diagnóstico completo do sistema. Uma outra abordagem para diagnóstico em nível de sistema é chamada de *diagnóstico adaptativo* [13]. Nesta abordagem, o diagnóstico é realizado em rodadas de testes (*testing rounds*) e, ao invés de determinar um conjunto fixo de testes, as unidades determinam quais testes devem ser executados baseadas nos resultados de testes

anteriores.

Um algoritmo adaptativo é apresentado em [13]. Tal algoritmo assume um sistema S de N unidades com no máximo t unidades falhas, e considera também o modelo de invalidação simétrica de testes. Inicialmente o processo de diagnóstico encontra uma unidade sem-falha u_i no sistema; em seguida, u_i é utilizada como testadora das demais unidades. Como é assumido o modelo de invalidação simétrica de testes, u_i irá retornar o estado correto de todos os demais nodos. Ainda em [13], é provado que o algoritmo é capaz de diagnosticar todas as unidades falhas do sistema com no máximo $N + 2t - 2$ testes.

3.1.4 Diagnóstico Distribuído

Tanto no diagnóstico adaptativo como no modelo PMC, a síndrome é coletada e analisada por uma unidade central ao sistema, a qual determina o estado das demais unidades. A existência desta unidade central pode representar dificuldades para a implementação destas técnicas de diagnóstico em certo ambientes, como em redes *ad hoc*. Diagnóstico distribuído em nível de sistema é uma outra abordagem de diagnóstico, a qual elimina a necessidade de uma unidade central. Nesta abordagem as próprias unidades do sistema são responsáveis pelo diagnóstico. Cada unidade armazena os estados de todas as demais unidades do sistema. Os resultados dos testes realizados pelas unidades são compartilhados entre os demais nodos permitindo um diagnóstico de todo o sistema, também chamado de rede.

O primeiro algoritmo de diagnóstico distribuído foi apresentado por Kuhl e Reddy [38]. O algoritmo, chamado *SELF*, embora seja completamente distribuído, é não-adaptativo, ou seja, cada unidade tem um conjunto fixo de testes a ser executado. Posteriormente Hosseini et al. [14] apresentaram o algoritmo *NEW-SELF*, baseado no algoritmo *SELF*. O algoritmo *NEW-SELF* também é não-adaptativo, porém permite que nodos falhos retornem ao sistema após serem reparados.

O algoritmo *Adaptive Distributed System-level Diagnosis* (ADSD), *Adaptive*-DSD, introduzido por Bianchini e Buskens [15, 39], é, ao mesmo tempo, adaptativo e completa-

mente distribuído. Os testes são executados em rodadas de testes. Uma rodada de testes é definida como o período necessário para que todas as unidades sem-falha testem pelo menos outra unidade sem-falha. Se existirem pelo menos duas unidades sem-falha no sistema, ao término de uma rodada de testes, o grafo que representa o conjunto de testes executados tem a forma de um anel. O algoritmo ADSD é executado em cada nodo em intervalos de testes, e a cada vez que o algoritmo é executado por um nodo sem-falha, este nodo executa testes sobre as demais unidades do sistema, até que uma unidade sem-falha seja encontrada, ou até que todas as unidades sejam testadas como falhas. Ao testar uma unidade sem-falha, o nodo testador obtém informações da unidade sem-falha para realizar o diagnóstico.

Supondo um sistema com N unidades u_1 , u_2 , ... , u_N, o algoritmo ADSD executa testes de forma sequencial, ou seja, o nodo u_i testa seu sucessor u_{i+1}, e assim sucessivamente, até que um nodo sem-falha seja encontrado. Para que a forma de anel seja mantida, a identificação dos nodos do sistema é feita de forma circular, ou seja, o sucessor do nodo u_N é u_1. O algoritmo ADSD tem uma latência de N rodadas de testes. Latência é o número de rodadas de teste necessárias para que todos os nodos tenham informações atualizadas de todos os demais nodos.

Para que uma menor latência seja obtida, Duarte et al. [16] propõem um novo algoritmo de diagnóstico distribuído, *Hierarchical Adaptive Distributed System-Level Diagnosis* (Hi-ADSD). O algoritmo Hi-ADSD, além de ser distribuído e adaptativo como o ADSD, utiliza uma organização hierárquica de testes, baseada em *clusters*. Um *cluster* é um conjunto de nodos. O tamanho de um *cluster*, ou seja, seu número de nodos, é sempre uma potência de 2.

Um nodo realiza testes, primeiramente, sobre um *cluster* de tamanho 1, e nas próximas rodadas o tamanho dos *clusters* aumentam progressivamente em potências de 2 até atingirem o tamanho de $\frac{N}{2}$, ou $2^{\log_2 N-1}$ nodos. Ao executar testes sobre um nodo sem-falha presente em um *cluster*, o nodo testador recebe informações de diagnóstico de todo o *cluster* ao qual o nodo testado pertence.

Em [16], Duarte et al. provam que o algoritmo Hi-ADSD é capaz de realizar o di-

agnóstico completo do sistema em no máximo $\log^2 N$ rodadas de testes. Modificações para o algoritmo Hi-ADSD foram propostas: Hi-ADSD *with detours* [17] e Hi-ADSD *with timestamps* [18]. Apesar de terem o mesmo número máximo de testes, ambos apresentam latências menores que o algoritmo *Hi-ADSD*, e simulações mostram que o algoritmo Hi-ADSD *with timestamps* é em média quatro vezes mais rápido do que o algoritmo Hi-ADSD *with detours*.

Em [19] é proposto um modelo teórico chamado *Bounded Correctness* que permite definir parâmetros necessários para a correção de algoritmos de diagnóstico distribuído para ambientes com eventos dinâmicos. Dois algoritmos são propostos em [19]: *HeartbeatComplete* e *ForwardHeartbeat*. O primeiro assume uma rede completamente conectada, enquanto o segundo assume que a rede precisa apenas ser conexa, mas estipula também um número máximo de unidades falhas para um dado instante. É provado também que ambos os algoritmos alcançam as propriedades de *bounded correctness*.

Algoritmos de diagnóstico em nível de sistema para redes de topologia arbitrária também foram propostos: algoritmo de Bagchi e Hakimi [40], *Adapt* [41], RDZ [42], NBND [43, 44], DNC [45], *ForwardHeartbeat* [19] e DNR [46] são alguns exemplos.

3.2 Diagnóstico Baseado em Comparações

Diagnóstico em nível de sistema baseado em comparações é uma abordagem que considera comparações entre resultados de testes de uma ou mais unidades e não testes realizados diretamente sobre uma unidade [47]. Ou seja, a saída produzida por uma unidade após um estímulo é comparada à saída produzida por outra unidade após o mesmo estímulo. De acordo com os resultados de um número suficiente de comparações, é possível identificar unidades falhas.

O primeiro modelo de diagnóstico baseado em comparações foi proposto por Malek [4] em 1980. O modelo assume um sistema com N unidades onde é possível, após determinada entrada, a comparação das saídas produzidas, por alguns, ou por todos, os pares de unidades. Se, para uma mesma entrada, duas unidades produzem resultados diferentes, então ambas as unidades podem ser consideradas falhas. Mais especificamente, o modelo

assume que: (1) as saídas produzidas por duas unidades sem-falha que executam uma mesma tarefa são sempre idênticas; e (2) a saída produzida por uma unidade falha é sempre diferente de qualquer saída produzida por outra unidade, falha ou sem-falha.

Ainda no modelo proposto por Malek, uma asserção é feita: a existência de uma unidade externa confiável, que nunca falha, a qual realiza as comparações e o diagnóstico do sistema. Os resultados das possíveis comparações entre duas unidades são listadas na Tabela 3.1. O resultado *pass* indica que ambas as unidades estão sem-falha, enquanto que o resultado *fail* indica que pelo menos uma das unidades é falha. É possível perceber que quando há discrepância nos resultados das comparações, ou seja, pelo menos uma das unidades está falha, mais comparações são necessárias para que o diagnóstico seja efetuado com sucesso.

Unidade 1	Unidade 2	Resultado da Comparação
Sem-falha	Sem-falha	0 (*pass*)
Sem-falha	Falha	1 (*fail*)
Falha	Sem-falha	1 (*fail*)
Falha	Falha	1 (*fail*)

Tabela 3.1: Possibilidades de resultados de comparações realizadas entre duas unidades no modelo proposto por Malek [4].

Ainda neste modelo, é provado que, em um sistema com N unidades no qual a comparação entre qualquer par de unidades é possível, o número máximo de unidades falhas é igual a $N-2$ para que o diagnóstico seja correto, ou seja, a diagnosticabilidade do sistema é $N-2$ [47].

Um modelo de diagnóstico semelhante ao apresentado por Malek foi proposto por Chwa e Hakimi [5] em 1981. A diferença principal entre os modelos ocorre com relação aos testes realizados entre duas unidades falhas. Enquanto no modelo sugerido por Malek quando duas unidades falhas executam uma mesma tarefa, elas sempre apresentam resultados diferentes, no modelo proposto por Chwa e Hakimi duas unidades falhas podem apresentar resultados iguais para uma mesma tarefa. Os possíveis resultados das comparações no modelo de Chwa e Hakimi são apresentados na Tabela 3.2.

Outro modelo de diagnóstico baseado em comparação foi proposto por Maeng e Malek

Unidade 1	Unidade 2	Resultado da Comparação
Sem-falha	Sem-falha	0 (*pass*)
Sem-falha	Falha	1 (*fail*)
Falha	Sem-falha	1 (*fail*)
Falha	Falha	0 ou 1

Tabela 3.2: Possibilidades de resultados de comparações realizadas entre duas unidades no modelo proposto por Chwa e Hakimi [5].

[6], projetado inicialmente para sistemas compostos por um conjunto de processadores homogêneos. Neste modelo, conhecido como modelo MM, as unidades do sistema são também unidades comparadoras, ou comparadores, ou seja, entidades que realizam o processo de comparação entre duas unidades do sistema. Desta forma, as unidades realizam comparações entre si, e então enviam os resultados para a unidade externa que conclui o processo de diagnóstico.

O sistema considerado pelo modelo MM é representado por um grafo $G = (V, E)$ onde V é o conjunto de unidades e E é o conjunto de enlaces de comunicação existentes entre as unidades. Assim, uma unidade k é um comparador das unidades i e j somente se $(k, i) \in E$ e $(k, j) \in E$, sendo $k \neq i$ e $k \neq j$. Unidades comparadoras também podem apresentar falhas. Desta forma, para que o diagnóstico seja correto, um mesmo par de unidades i e j pode ser testado por mais de um comparador. A Tabela 3.3 apresenta as possibilidades de resultados das comparações realizadas no modelo MM.

Comparador	Unidade 1	Unidade 2	Resultado da Comparação
Sem-falha	Sem-falha	Sem-falha	0 (*pass*)
Sem-falha	Sem-falha	Falha	1 (*fail*)
Sem-falha	Falha	Sem-falha	1 (*fail*)
Sem-falha	Falha	Falha	1 (*fail*)
Falha	Sem-falha	Sem-falha	0 ou 1
Falha	Sem-falha	Falha	0 ou 1
Falha	Falha	Sem-falha	0 ou 1
Falha	Falha	Falha	0 ou 1

Tabela 3.3: Possibilidades de resultados de comparações realizadas no modelo MM [6].

As principais asserções do modelo MM são [47]: (1) toda falha é permanente, ou seja, unidades não se recuperam de estados falhos; (2) uma comparação realizada por uma unidade falha tem resultado arbitrário; (3) duas unidades falhas que executam uma

mesma tarefa sempre produzem saídas diferentes; (4) cada unidade falha gera uma saída incorreta para toda e qualquer tarefa; desta forma, qualquer comparação realizada entre uma unidade falha e uma outra unidade qualquer sempre apresenta discrepância; e (5) existe um limite máximo t de unidades do sistema que podem estar falhas, o que permite o diagnóstico correto do sistema.

O modelo MM possibilita um caso especial no qual cada unidade realiza a comparação entre todos os pares possíveis dentre suas unidades vizinhas. Este caso especial é tratado como outro modelo, conhecido como modelo MM*.

Sengupta e Dahbura [48] propõem uma generalização para o modelo MM na qual é permitido que comparadores sejam uma das unidades comparadas, ou seja, uma unidade pode comparar seu próprio comportamento com outra unidade do sistema. A comparação realizada entre o próprio comparador e outra unidade do sistema pode ser considerada equivalente ao teste realizado no modelo PMC. Desta forma, o modelo apresentado em [48] é também uma generalização do modelo PMC. O mesmo trabalho apresenta também estudos sobre a diagnosticabilidade do modelo MM e um algoritmo de diagnóstico $O(n^5)$ para o modelo MM*.

Outros modelos apresentam abordagens diferentes para o diagnóstico baseado em comparações. Diagnóstico probabilístico [49], onde os modelos assumem a probabilidade de uma unidade apresentar comportamento incorreto é um exemplo. Outra abordagem é o diagnóstico distribuído. Em [50] é apresentado um modelo de diagnóstico distribuído baseado em comparação através da disseminação de resultados por *broadcast* de comparações entre pares de unidades do sistema. Outro modelo distribuído, porém sem o uso de disseminação de resultados por *broadcast* é apresentado por [51] e [52].

Modelos de diagnóstico baseados em comparações permitem um grande número de aplicações. O campo de sistemas multiprocessados, por exemplo, utiliza o diagnóstico baseado em comparações em redes de processadores, ou em processadores com mais de um núcleo [53]. O número de núcleos presentes em um único chip têm crescido muito, e acredita-se que num futuro próximo esse número chegue a centenas de núcleos, necessitando assim de um sistema de diagnóstico capaz de identificar unidades falhas eficiente-

mente. Outras aplicação para diagnóstico baseado em comparações são a identificação de unidades falhas em redes *ad hoc* [54, 9], verificar a presença de nodos maliciosos em sistemas de computação em grade [55] ou ainda, verificar a integridade de dados replicados em sistemas distribuídos [52].

3.3 Tendências Atuais em Diagnóstico em Nível de Sistema

Pesquisas recentes apresentam novas abordagens, aplicações e algoritmos de diagnóstico em nível de sistema.

Estudos da diagnosticabilidade de diferentes topologias têm sido realizados. Em [56] são apresentados resultados sobre a diagnosticabilidade da topologia conhecida como *hypermesh*, baseada em hipergrafos. Um algoritmo de diagnóstico para a topologia *hypermesh* também é proposto. O trabalho apresentado em [57] trata da diagnosticabilidade de redes com topologias baseadas em grafos do tipo *star-pyramid*. Outro trabalho, [58], apresenta um estudo da diagnosticabilidade de grafos lineares congruentes DCC [59].

Elhadef propõe em [60] o uso de redes neurais no diagnóstico de sistemas compostos por unidades heterogêneas. O diagnóstico proposto utiliza uma síndrome gerada a partir do modelo de comparação assimétrico, introduzido por Malek em [4], para encontrar as unidades falhas presentes no sistema. A solução de Elhadef se baseia em redes neurais do tipo *perceptron* [61]. A rede neural é primeiramente testada usando como entrada várias síndromes com conjunto de unidades falhas conhecido. Após a fase de treinamento, a rede é então utilizada na avaliação de síndromes. O trabalho mostra também que o algoritmo proposto apresenta resultados promissores, atingindo a taxa de 100% de acertos nos testes realizados pelo estudo. Elhadef ainda compara sua abordagem com trabalhos baseados em algoritmos evolutivos, apresentando resultados superiores a estes algoritmos.

Algoritmos evolutivos [62] têm sido utilizados como ferramenta para a realização de diagnóstico de sistemas. Por exemplo, em [63] é proposto o uso de *sistema imunológicos artificiais* para o diagnóstico de sistemas.

Em [64] é proposto um modelo de diagnóstico em nível de sistema para grades computacionais (*grid computing*). Este modelo utiliza estratégias de votação e de *honeypots*

para encontrar unidades maliciosas na grade. Unidades maliciosas são componentes da grade que atuam de forma a denegrir o desempenho e/ou o funcionamento do sistema. O trabalho ainda demonstra resultados do modelo apresentado. Em alguns testes, o modelo alcançou a taxa de 99,4% de *jobs* concluídos com sucesso na grade computacional, mesmo com a presença de nodos maliciosos.

3.4 Algoritmos de Diagnóstico para Redes de Sensores

O recente aumento no interesse por redes de sensores e suas aplicações tem implicado em várias pesquisas desenvolvidas para o tema, inclusive pesquisas sobre diagnóstico em nível de sistema específicos para redes de sensores.

Chessa e Santi, apresentam em [9] uma estratégia de testes baseada em comparações para redes *ad hoc*. A estratégia explora o paradigma de comunicação de um-para-muitos (*one-to-many*), característico das redes *ad hoc*. O diagnóstico é realizado através de testes entre pares de unidades com vizinhos em comum. Os resultados destes testes são repassados pelas unidades da rede até que todas tenham informações atualizadas do diagnóstico. Os autores propõem, ainda, duas implementações para o modelo apresentado. A primeira implementação assume que a topologia da rede permanece inalterada durante todo o processo de diagnóstico; a segunda permite a movimentação livre das unidades. Simulações realizadas mostram que a mobilidade das unidades dificulta consideravelmente a tarefa de diagnóstico da rede.

Em outro trabalho Chessa e Santi propõem um protocolo de diagnóstico para redes de sensores. Este protocolo, chamado de *WSNDiag* [65], tem como objetivo diagnosticar falhas do tipo *crash*. Neste contexto, uma unidade falha não realiza nenhum tipo de comunicação com o restante da rede e mantém o estado de falha permanentemente. No protocolo *WSNDiag*, o processo de identificação de unidades falhas se inicia após uma unidade sem-falha do sistema ou uma unidade externa solicitar o diagnóstico. Desta forma, o diagnóstico funciona por demanda, resultando em economia de energia para a rede. O *WSNDiag* é capaz de realizar o diagnóstico correto com até t unidades falhas presentes na rede, sendo $t < k(G)$, onde $k(G)$ é a conectividade da rede.

No protocolo *WSNDiag* uma unidade sem-falha inicia o processo de diagnóstico através do envio de mensagens do tipo *I'm alive*, ou IMA. Tais mensagens são repassadas para todos os nodos da rede, que repetem o processo, gerando assim uma estrutura em árvore de mensagens. De acordo com as respostas obtidas, os nodos da rede são capazes de definir quais de seus vizinhos estão falhos. A informação dos nodos vizinhos falhos são repassadas para o nodo "pai" na árvore de mensagens IMA, alcançando, assim, o nodo que iniciou o processo de diagnóstico. Uma vez tendo o conjunto de todas as unidades falhas do sistema, essa informação é repassada para toda a rede.

Em [66] é apresentado um esquema adaptativo e distribuído para detecção de falhas em redes de sensores. Neste esquema cada sensor avalia suas próprias leituras (ou informações de medições do ambiente de sensoriamento) através da comparação com as leituras de seus vizinhos. Se o número de vizinhos com a mesma leitura que o próprio nodo for igual ou superior a um certo limiar θ, então o nodo se auto declara como sem-falha. O algoritmo ainda é capaz de adaptar o limiar θ de forma a melhorar a acurácia da detecção de falhas no decorrer das rodadas de testes. O trabalha ainda apresenta o uso de redundância de tempo para que falhas transientes também sejam detectadas pelo algoritmo. O trabalho também mostra que o algoritmo obtém uma taxa de detecção de falhas satisfatória.

Outra abordagem muito utilizada em redes de sensores é a organização das unidades da rede em *clusters*, ou seja, a rede é dividida em vários grupos lógicos de nodos. Em [67] é proposta uma abordagem para detecção de nodos com falhas do tipo *data fault* em redes de sensores. Unidades com esse tipo de falha transmitem informações com dados incorretos. Mais precisamente, em redes de sensores, esse tipo de falha se caracteriza quando nodos informam medições e análises do ambiente de forma incorreta, por exemplo, sensores de temperatura retornam valores indicando calor para ambientes frios. Em [67, 68], falhas do tipo *data fault* são categorizadas em três diferentes classes: (1) curta (*short*): dentre as informações retornadas por um sensor em um dado período de tempo, apenas uma delas contem valores incorretos; (2) ruído (*noise*): dentre as informações retornadas por um sensor em um período de tempo, uma pequena sequência delas contêm valores incorretos; e (3) constante (*constant*): todas as informações retornadas por um sensor contêm valores

incorretos.

A abordagem apresentada em [67] utiliza conceitos de confiança e reputação entre as unidades da rede. A rede é dividida em *clusters*, e cada *cluster* possui uma unidade nomeada *cluster head* (CH). As unidades pertencentes a um mesmo *cluster* compartilham e comparam entre si suas informações. A partir destas comparações cada unidade constrói um conjunto de possíveis vizinhos falhos, e repassa esse conjunto ao CH do *cluster*. Por sua vez, o CH verifica as unidades que tiveram um maior número de indicações de falhas para encontrar o conjunto de possíveis nodos falhos presentes no *cluster*, e o repassa ao *sink*. Os autores afirmam, ainda, que esta abordagem proporciona um diagnóstico com baixo consumo de energia.

Em [69] é proposta uma técnica com consumo eficiente de energia e baseada em *clusters*, para a detecção de falhas em redes de sensores. A técnica tem como principal objetivo evitar a perda de desempenho da rede, causada pela redução de conectividade gerada por nodos falhos. As falhas detectadas são, principalmente, as causadas por falta de energia. Desta forma, a rede pode localizar sensores com baixos níveis de energia, e antes de se tornarem falhos e causarem danos ao desempenho da rede, a organização dos *clusters* pode ser recuperada ou reajustada.

CAPÍTULO 4

ESTRATÉGIAS DE ASSINALAMENTO DE TESTES

Aplicações de redes de sensores necessitam que, mediante detecção de algum evento pré-determinado, sensores informem à unidade central (*sink*) detalhes sobre o fenômeno. Estas informações são transmitidas através de mensagens de alarme. Ao receber um alarme, o *sink* repassa informações ao usuário para que, se necessário, medidas sejam tomadas. Sensores estão sujeitos a falhas, as quais podem causar sensoriamento incorreto. Desta forma, falhas podem ocasionar mensagens de alarme incorretas. Em algumas aplicações, a informação incorreta sobre um fenômeno pode ocasionar tomada de medidas desnecessárias que, muitas vezes, geram altos custos [2, 1]. Desta forma, a criação um sistema capaz de avaliar se as mensagens de alarme recebidas pelo *sink* são, ou não, verdadeiras, pode evitar a tomada de tais medidas desnecessárias.

Considere uma rede de sensores onde um conjunto de sensores próximos geograficamente enviam sinais de alarme ao *sink*, informando a detecção de determinado fenômeno. Para garantir que os alarmes gerados não são falsos, ou seja, que o fenômeno realmente ocorreu, o *sink* solicita a execução de testes entre os sensores presentes na região que contém os nodos que reportaram os alarmes. O resultado dos testes é coletado pelo *sink* que, então, os submete a um algoritmo de diagnóstico com o objetivo de detectar sensores falhos. As falhas detectadas são relacionadas com a capacidade de sensoriamento dos sensores. Qualquer fato que gere um sensoriamento incorreto é entendido como um sensor falho. Defeitos na unidade de processamento, na unidade de transmissão ou na própria unidade de sensoriamento são exemplos de problemas capazes de danificar a capacidade de sensoriamento do sensor [1].

A abordagem de diagnóstico em nível de sistema permite que, através de testes executados entre as próprias unidades do sistema, unidades falhas sejam identificadas. No modelo PMC, testes são dependentes da aplicação, mas de forma geral são um conjunto

de estímulos enviados de uma unidade a outra. O conjunto de testes executados entre os sensores é representado por um grafo direcionado chamado *grafo de testes*. Para que sensores falhos sejam identificados corretamente o grafo de testes deve possuir algumas propriedades topológicas que permitam que uma *diagnosticabilidade* suficiente seja alcançada. Sem a garantia de que o grafo de testes possua diagnosticabilidade suficiente, o algoritmo de diagnóstico não é capaz de encontrar as unidades falhas em todos os casos. Um grafo de testes é chamado, também, de assinalamento de testes.

Este trabalho propõe e compara três estratégias de testes para um algoritmo de diagnóstico em nível de sistema para rede de sensores sem fio: TAWR [70] (*Test Assignment Without Reciprocal Tests*), EETA [71] (*Energy-Efficient Test Assignment*) e ODTA (*Optimal Design Test Assignment*). Tais estratégias tem como objetivo gerar, de forma eficiente, assinalamentos de testes com diagnosticabilidade suficiente para validar um conjunto de alarmes. As estratégias apresentadas neste trabalho garantem que, para um alarme gerado por t sensores, um grafo de testes no mínimo t-diagnosticável seja gerado. As estratégias visam, também, a economia de energia durante o processo de diagnóstico.

Este Capítulo apresenta em detalhes as estratégias de testes e está organizado da seguinte forma: a Seção 4.1 descreve o modelo do sistema considerado. As Seções 4.2, 4.3 e 4.4 descrevem, respectivamente, os modelos de falhas, de testes e de energia utilizados no decorrer deste trabalho. Detalhes sobre assinalamentos de testes e suas propriedades são tratados na Seção 4.5. Por fim, a Seção 4.6 apresenta as estratégias de testes propostas.

4.1 Modelo de Sistema

No modelo utilizado por este trabalho é considerada uma rede de sensores sem fio onde nodos são depositados em uma área de sensoriamento seguindo uma distribuição uniforme, apresentando uma cobertura completa da área.

Assume-se que cada nodo tem conhecimento de sua própria posição geográfica. A aquisição das coordenadas geográficas de cada sensor pode ser realizada através de dispositivos GPS ou de técnicas de localização, como as apresentadas no Capítulo 2. Ainda, cada sensor da rede repassa informações de sua posição geográfica para seus vizinhos ime-

diatos, ou seja, nodos que estão no raio de transmissão do sensor. Assume-se, também, que o *sink* conhece a posição geográfica de todos os sensores da rede.

A rede de sensores utilizada neste trabalho é modelada como o grafo $G = (V, E)$, onde cada vértice presente em V representa um nodo da rede, e uma aresta (v_i, v_j) existe em E se e somente se v_i está no raio de transmissão do sensor v_j, e vice versa.

Os nodos presentes na rede realizam o sensoriamento do ambiente com o objetivo de gerar alarmes quando determinado fenômeno é percebido. Por exemplo, sensores capazes de detectar níveis de fumaça, de monitorar parâmetros físico-químicos da água ou propriedades do ar, ao perceberem níveis superiores a um certo limiar, reportam mensagens de alarme. As mensagens são enviadas até o *sink*, que então, informa o usuário remoto do evento. Assume-se que um conjunto de sensores $T \subset V$, de cardinalidade t, reporta mensagens de alarme com relação a um fenômeno pré-determinado. Os alarmes gerados dentro de um certo período de tempo e em uma certa proximidade geográfica (dados que são determinados pela aplicação) são considerados informações sobre um mesmo evento, formando, assim, o conjunto T. O conjunto de todos os sensores escolhidos para participar do processo de diagnóstico é denotado como V_D, e a cardinalidade de V_D como n. Claramente $T \subset V_D$. A definição do conjunto V_D é um dos objetivos das estratégias de testes.

Para garantir que os alarmes gerados são corretos, ou seja, o fenômeno informado realmente ocorreu, o *sink* solicita uma série de testes entre os nodos presentes em V_D. Segundo o modelo PMC, utilizado aqui, um teste (v_i, v_j) é, de forma geral, um conjunto de estímulos de entrada produzido pelo sensor testador v_i e enviado ao sensor testado v_j. O sensor v_j, por sua vez, responde ao conjunto de estímulos e envia o conjunto de resultados ao sensor v_i. Por fim, v_i compara os resultados gerados por v_j com valores esperados produzindo o resultado do teste, que é composto por um valor binário: 0 se os resultados conferem (nodo v_j é testado como sem-falha), e 1 caso contrário (nodo v_j é testado como falho). Conforme as asserções do modelo PMC, um teste executado por uma unidade sem-falha apresenta resultado sempre confiável; por outro lado, se o teste é executado por uma unidade falha, o resultado é arbitrário.

O conjunto de todos os resultados dos testes executados é enviado ao *sink*, que então, realiza a decodificação da síndrome através de um algoritmo de diagnóstico.

O objetivo deste trabalho é definir assinalamentos de testes entre os nodos presentes em V_D capazes de gerar um grafo $D = (V_D, E_D)$ de diagnosticabilidade maior ou igual a t, onde uma aresta (v_i, v_j) existe em E_D se e somente se v_i realiza testes sobre v_j.

4.2 Modelo de Falhas

Para diagnóstico em nível de sistema, o modelo de falhas é uma descrição dos resultados dos testes dados os estados das unidades testadora e testada [3]. Um procedimento de diagnóstico pode levar em conta possíveis classes de falhas durante o processo de testes. Uma classificação amplamente utilizada baseada no domínio temporal é apresentada na Figura 4.1. Falhas do tipo *fail-stop* ocorrem quando uma unidade cessa sua operação e informa outras unidades sobre a falha [72]. Falhas do tipo *crash* ocorrem quando uma unidade deixa de funcionar e de responder a estímulos. Falhas de omissão ocorrem quando uma unidade falha em iniciar uma tarefa ou cumpri-la completamente [73]. Falhas temporais ocorrem quando uma unidade completa uma tarefa antes ou depois do tempo estabelecido, ou ainda, jamais conclui a tarefa [73]. Falhas de computação incorreta ocorrem quando uma unidade não apresenta resultados corretos para determinados estímulos [74]. Falhas bizantinas autenticadas são falhas arbitrárias ou maliciosas [75]. E, por fim, falhas bizantinas são o conjunto de todas as falhas possíveis no modelo do sistema [75]. Esta classe pode ser considerada o conjunto universal de falhas. As classes mais básicas de falha, *crash*, omissão e falhas temporais, são problemas que ocorrem no domínio temporal e são detectáveis, também, naquele domínio [3]. A classe de falha de computação incorreta, por sua vez, é uma superclasse das falhas *crash*, omissão e falhas temporais e é uma subclasse das falhas bizantinas. A classe de falha de computação incorreta é mais restrita do que as falhas bizantinas, uma vez que a falha de computação incorreta é consistente para todos os observadores externos [74].

Ainda, o modelo de falhas do domínio temporal é ortogonal às técnicas de falhas do domínio de dados. No domínio de dados, isto é, quando não é considerado o tempo do

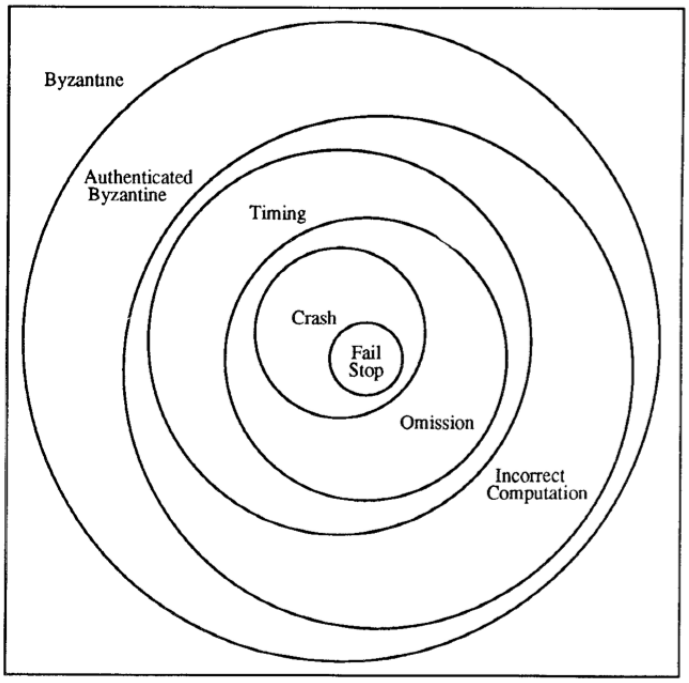


Figura 4.1: Uma classificação de falhas [3].

estímulo de entrada, a única forma de uma unidade apresentar falhas é se tal unidade produzir respostas incorretas. Existem diversas formas de interpretação do domínio de falhas de dados na literatura. Em [68], Sharma et al. introduzem uma classificação para falhas de sensoriamento em três tipos: falhas *short*, *noise* e *constant*. Falhas do tipo *short* se referem a uma alteração brusca no valor medido entre duas leituras consecutivas. Falhas do tipo *noise* ocorrem quando a variância nas leituras do sensor aumenta. Diferentemente de falhas do tipo *short* que afetam apenas uma leitura, falhas do tipo *noise* provocam alterações em um conjunto de leituras consecutivas. Por fim, falhas do tipo *constant* são caracterizadas quando um sensor reporta um valor constante de leitura durante um grande período de tempo. Este valor ou está muito acima ou muito abaixo do esperado para o comportamento de um sensor com funcionamento considerado "normal" e não mostra correlação com o fenômeno subjacente sendo monitorado.

Além disso, falhas podem ser classificadas como transientes, intermitentes ou permanentes [3]. Falhas transientes são causadas por eventos que surgem no ambiente do sistema e não implicam que o sistema é falho. Uma falha intermitente, ou *soft fault*, se origina no interior do sistema quando *software* ou *hardware* estão falhos. Por sua natureza, uma

falha intermitente não ocorrerá consistentemente, o que torna o seu diagnóstico um evento probabilístico. Falhas permanentes, ou *hard fault*, são falhas de *software* ou *hardware* que sempre produzem erro quando são completamente exercitadas. Para Elhadef et al. [76] falhas em sensores são classificadas como *hard* (*crash*, *fail-stop* e *fail-silent*) ou *soft*. Se o sensor falho não é capaz de se comunicar com seus vizinhos, a falha é classificada como *hard*. Por outro lado, se a unidade falha continua a se comunicar porém a funcionar com comportamento corrompido, a falha é dita *soft*.

Estudos sobre aplicações reais com redes de sensores têm demonstrado que falhas de sensoriamento são relativamente comuns, causando leituras incorretas realizadas pelos sensores [68]. Questões como falhas no desenvolvimento de *hardware*, erros de calibragem ou baixos níveis de bateria têm se mostrado como causadoras de leituras incorretas.

Neste trabalho são consideradas falhas de computação incorreta, visto que um sensor falho no domínio de dados exibe resultados incorretos quando procedimentos de testes são exercitados. Além disso, as falhas podem ser consideradas do tipo *soft* segundo [76], uma vez que o sensor é capaz de comunicar-se ainda que o sensoriamento seja incorreto.

4.3 Modelo de Testes

Além do teste clássico do modelo PMC, onde a unidade testadora envia estímulos para a unidade testada e analisa o resultado obtido, outra modalidade de teste pode ser utilizada. Trata-se dos *self-tests* [3], em que uma unidade executa a rotina de testes sobre si mesma e o teste de uma unidade sobre a outra torna-se uma requisição sobre o estado da última.

Além disso, considerando que os sensores de T serão testados por sensores relativamente próximos, pertencentes a $V_D - T$, estes deverão ser capazes de avaliar a leitura obtida pelos sensores de T. Desta forma, se a leitura estiver fora de um certo limiar Θ, dependente da aplicação e da distância entre os sensores, assume-se que o sensor é falho.

4.4 Modelo de Energia

Para estimar o consumo de energia de um determinado assinalamento de testes, é considerado neste trabalho o modelo de atenuação de sinal *one-slop* [77], comumente utilizado em comunicações sem fio. Este modelo assume uma dependência linear entre o *path loss* (dB), ou atenuação do sinal, e o logaritmo da distância d entre o transmissor e o receptor, como mostrado na Equação 4.1:

$$L(d) \mid_{dB} = l_0 + 10\alpha \log_{10}(d) \tag{4.1}$$

onde l_0 é a atenuação do sinal (*path loss*) em uma distância de referência equivalente a 1 metro, e α é o fator de atenuação (*pathloss exponent*). Em geral, para garantir a comunicação entre um transmissor x e um receptor y posicionados a uma distância (em metros) d entre si, é necessário que o pacote enviado por x alcance y com um nível maior de energia do que a sensibilidade do receptor. Em outras palavras, seja E_t a potência de transmissão de x, E_r a potência do sinal ao alcançar o receptor (onde E_r depende de E_t e da distância d) e seja E_m a sensibilidade do receptor, é necessário que $E_r > E_m$.

Sabendo que $L(d)$ é igual à diferença em decibéis da potência do sinal no transmissor e sua potência ao alcançar o receptor [78], pela Equação 4.1, temos:

$$10 \log_{10} \left(\frac{E_t}{E_r} \right) = l_0 + 10\alpha \log_{10} d \tag{4.2}$$

Desenvolvendo temos:

$$E_r = \frac{E_t}{10^{\left(\frac{l_0}{10} + \alpha \log_{10} d \right)}} \tag{4.3}$$

Adicionando $E_r > E_m$ na Equação 4.3 obtém-se que o minimo de potência de transmissão E_t no transmissor que garante que o pacote alcance o receptor com o nível de energia necessário é:

$$E_t = E_m 10^{\left(\frac{l_0}{10} + \alpha \log_{10} d \right)} \tag{4.4}$$

Portanto, a Equação 4.4 depende da distância d, da sensibilidade do receptor e dos parâmetros l_0 e α. Para estes parâmetros são utilizados valores típicos para as simulações [78] (neste trabalho utilizamos $l_0 = 10$ e $\alpha = 3$), enquanto E_m depende do hardware dos sensores. Desenvolvendo a Equação 4.4 temos:

$$E_t = E_m 10^{\frac{l_0}{10}} d^{\alpha} \tag{4.5}$$

A partir da Equação 4.5 observa-se que a energia transmitida cresce polinomialmente com a distância d, com um expoente igual a α. Desta forma, as estimativas de custos energéticos feitas pelas estratégias de testes apresentadas neste trabalho baseiam-se exclusivamente na distância geográfica entre os sensores. Assim, para $l_0 = 10$ e $\alpha = 3$, temos:

$$E_t = E_m 10 d^3 \tag{4.6}$$

Pelo fato de E_m depender do hardware utilizado, e ser de valor muito pequeno, nota-se que a energia gasta é definida, em grande parte, pelo cubo da distância entre origem e destino. Assim, estabelecemos que uma unidade de energia, ou u.e., é igual a $10E_m$. Desta forma, definimos que a energia gasta para uma transmissão de d metros é igual a d^3 unidades de energia. Sabendo que o consumo de um teste será xd^3 u.e., onde x é uma constante que define o tamanho ou número de mensagens de um teste, utilizamos, neste trabalho, apenas o valor d^3 nas comparações entre as estratégias.

4.4.1 Custos de Energia Associados às Estratégias de Testes

Para o restante deste trabalho, são definidas algumas variáveis de consumo energético:

- custo energético de teste, $C_{i,j}$;
- custo energético total do assinalamento D, $C_T(D)$;
- custo energético médio por sensor do assinalamento D, $C_M(D)$;
- custo energético máximo por sensor do assinalamento D, $C_{max}(D)$.

Custo, ou consumo, energético de teste, $C_{i,j}$, é definido como o custo energético gasto pelos sensores v_i e v_j caso o sensor v_i execute um teste sobre o sensor v_j.

Custo energético total do assinalamento de testes D, $C_T(D)$, é igual a soma do custo energético de todos os testes executados pelos sensores durante o processo de diagnóstico.

Custo energético médio por sensor do assinalamento de testes D, $C_M(D)$, é definido como a média de custo energético entre os sensores de D, ou seja, $C_M(D) = \frac{C_T(D)}{n}$.

Custo energético máximo por sensor do assinalamento de testes D, $C_{max}(D)$ é, por sua vez, igual ao consumo de energia apresentado pelo sensor cujo somatório de todos os custos energéticos de teste executados por ele é o maior entre os sensores de D.

4.5 Assinalamentos de Testes

Conforme apresentado no Capítulo 3, para o propósito de diagnóstico em nível de sistema, um sistema S é dito t-diagnosticável se com até t unidades falhas presentes em S, a unidade central é capaz de realizar o diagnóstico. O modelo PMC, descrito em [8], introduz condições para que um sistema seja t-diagnosticável: $(c1)$ o número N de nodos no sistema deve satisfazer $N \geq 2t + 1$, onde t é a diagnosticabilidade do sistema; e $(c2)$ cada unidade deve ser testada por pelo menos t outras unidades. Hakimi e Amin demonstram em [12] que as condições $(c1)$ e $(c2)$ são necessárias e suficientes para que um sistema seja t-diagnosticável se não houverem testes mútuos entre as unidades. Para casos onde existem testes mútuos, uma terceira condição deve ser satisfeita, para a qual um corolário é dado: seja G um grafo direcionado que representa o sistema S, e $k(G)$ a conectividade do grafo G; se $k(G) \geq t$, então S é t-diagnosticável.

Desta forma, a diagnosticabilidade do grafo D, gerado pelos assinalamentos de testes, depende, além de outros fatores, da presença ou não de testes recíprocos em D. Testes recíprocos existem se e somente se ambas as arestas (v_i, v_j) e (v_j, v_i) estão presentes no grafo D.

A estratégia de testes estabelece a forma como os sensores presentes no conjunto V_D devem coordenar ações para que os testes realizados atinjam seu objetivo. Para isso a região que compreende os sensores de V_D deve ser delimitada. Se esta região

for demasiadamente grande, muitos sensores terão que realizar testes, causando um alto consumo de energia na rede. Por outro lado, se a região for muito pequena, as condições de diagnosticabilidade podem não ser alcançadas por falta de sensores ou de testes. Assume-se que são selecionados para o conjunto $V_D - T$ sensores que não apresentam falhas no domínio de tempo.

4.5.1 Assinalamentos de Testes e Testes Recíprocos

Penrose apresenta em [79] conclusões sobre a relação entre a conectividade alcançada por grafos e seus graus mínimos. Penrose prova que quando n tende ao infinito, a probabilidade de que o grafo seja t-conexo tende a 1 quando a probabilidade de qualquer vértice do grafo apresentar grau mínimo igual a t também tende a 1.

Os resultados apresentados por Penrose sugerem que se a densidade da rede for suficientemente alta então a condição ($c3$) é alcançada. Embora tais resultados indiquem que assinalamentos de testes com testes recíprocos são possíveis, a quantidade de testes e, consequentemente, de energia utilizada, são maiores quando comparadas com estratégias sem testes recíprocos, onde o número de testes pode ser bastante menor. E, em certos ambientes com baixa densidade de sensores, a existência de um assinalamento de testes com testes recíprocos não é garantida.

As estratégias presentes neste trabalho são baseadas em assinalamentos sem testes recíprocos e na diminuição do gasto energético. Tal diminuição ocorre tanto pela minimização do número de sensores em V_D quanto pela escolha de sensores próximos geograficamente entre si para participarem do diagnóstico.

4.6 Estratégias de Testes

Nesta Seção são apresentadas três estratégias de testes capazes de gerar grafos de testes t-diagnosticáveis em redes de sensores sem fio. Todas as estratégias visam a economia no consumo de energia dos sensores durante o processo de diagnóstico.

4.6.1 TAWR (*Test Assignment Without Reciprocal Tests*)

A estratégia de testes chamada TAWR (*Test Assignment Wthout Reciprocal Tests*) visa a definição distribuída do assinalamento de testes. Ao receber mensagens de alarme o *sink* solicita que cada sensor de V_D defina quais sensores serão seus testadores.

A estratégia TAWR fundamenta-se na definição de uma região geográfica retangular R na qual os sensores de T e um número suficiente de outros sensores estão presentes. Essa região é dividida em 4 quadrantes adjacentes e de mesmo tamanho. Sensores presentes no mesmo quadrante não executam testes entre si. A estratégia de testes se baseia na solicitação de testes entre nodos de quadrantes diferentes. A Figura 4.2 apresenta a estratégia de particionamento de R e uma possível organização de testes entre os sensores, neste caso, para diagnosticabilidade igual a 2.

Cada quadrante possui apenas dois quadrantes vizinhos e, seguindo o sentido anti-horário, podemos nomeá-los quadrante predecessor e quadrante sucessor. Sensores de um quadrante solicitam testes de sensores presentes no quadrante sucessor, e são solicitados para executar testes em nodos presentes no quadrante predecessor. A Figura 4.3 apresenta um exemplo de raio de transmissão de um dos sensores da rede e seus vizinhos no quadrante sucessor aos quais poderá solicitar testes.

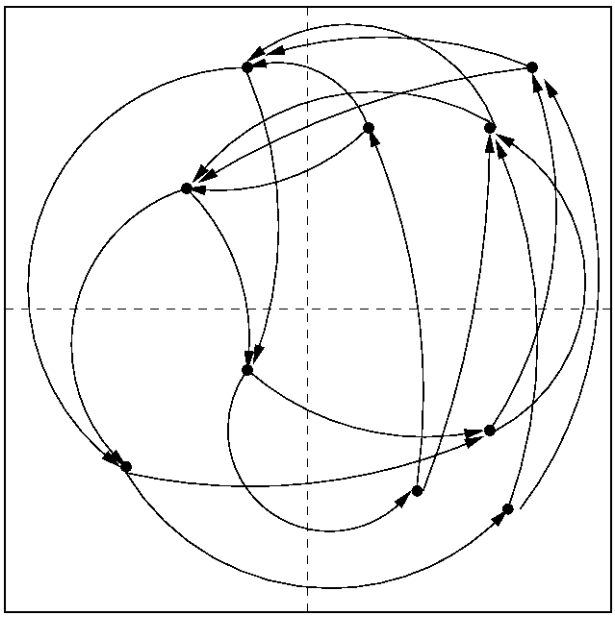


Figura 4.2: Região de testes dividida em quadrantes e uma possível organização de testes entre os nodos para diagnosticabilidade igual a 2.

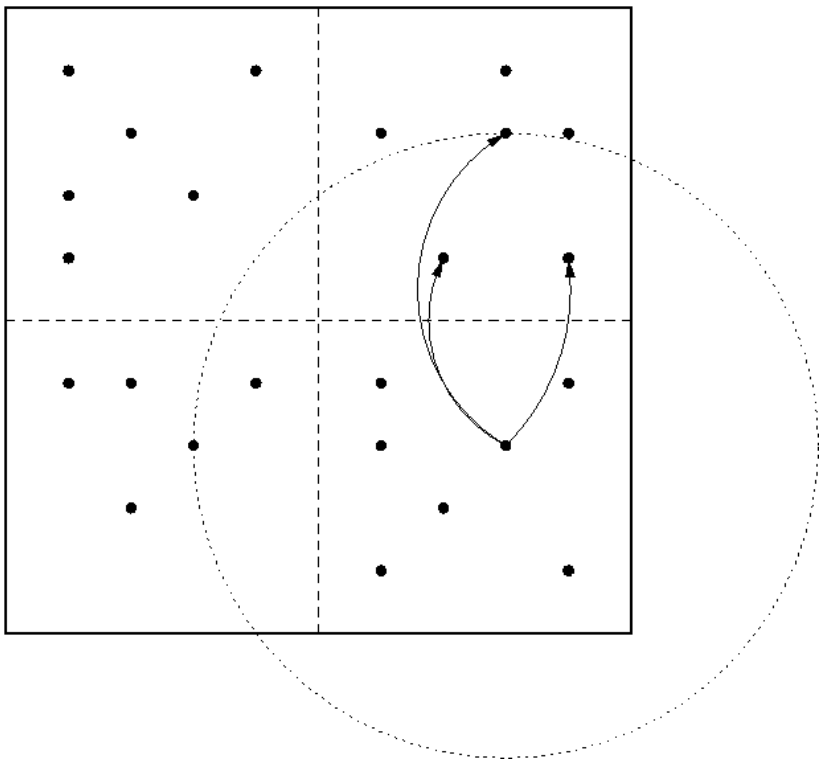


Figura 4.3: Nodo com seu raio de transmissão e seus possíveis testadores no quadrante sucessor.

Para que o grafo de testes D seja t-diagnosticável, as condições $(c1)$ e $(c2)$ devem ser satisfeitas. Para satisfazer $(c1)$ basta que $n \geq 2t + 1$. Para que $(c2)$ seja alcançada, é necessário que cada sensor encontre pelo menos t sensores testadores no quadrante sucessor.

Com o objetivo definir uma região R onde n seja suficientemente grande para possibilitar a diagnosticabilidade desejada, uma heurística é utilizada. Nesta heurística, a região R inicia-se como o menor retângulo que compreende os nodos de T, e então, é expandida até que cada quadrante possua, no mínimo, t sensores. A partir desta abordagem, as condições $(c1)$ e $(c2)$ são satisfeitas, pois cada quadrante tem no mínimo t sensores, garantindo que cada sensor pode encontrar t testadores, e $n \geq 2t + 1$, pois $4t \geq 2t + 1$. Ainda que no melhor caso o número mínimo de sensores utilizados seja $4t$, o processo de formação da região R, em geral, agrega um número maior de sensores para o diagnóstico.

4.6.1.1 Nodos de Borda

Para que a diagnosticabilidade desejada de D seja alcançada se faz necessário o tratamento dos nodos de borda. Nodos de borda são definidos como sensores para os quais nem todos os testadores pertencem à região R, ou seja, sensores próximos aos limites de R cujos raios de transmissão não alcançam pelo menos t sensores no quadrante sucessor.

O aumento do raio de transmissão dos sensores, apesar de gerar um consumo maior de energia e favorecer maior interferência nas comunicações, pode ser um alternativa para solucionar problemas de nodos de borda na estratégia TAWR.

Nesta estratégia os sensores não conhecem, a priori, o tamanho do raio de transmissão necessário para alcançar t testadores. Desta forma, um algoritmo de aumento progressivo do raio de transmissão de cada sensor é utilizado para que o aumento do consumo de energia seja melhor distribuído entre os sensores da rede. Cada sensor v_i presente em R executa o Algoritmo 1. Este algoritmo garante que um sensor alcance pelo menos t testadores e que todos seus t testadores sejam capazes de alcançá-lo também, possibilitando tanto o uso de testes bi-direcionais quanto uni-direcionais. O algoritmo ainda permite que o aumento dos raios de transmissão seja eficiente, visto que apenas os nodos que realmente necessitam de raios de transmissão maiores o realizarão.

Algoritmo 1: aumento progressivo de raio de transmissão

```
enquanto sensor v_i não encontra t testadores faça
    sensor v_i aumenta seu raio de transmissão
    para cada sensor v_j alcançável por v_i faça
        sensor v_i solicita testes para nodo v_j
        se sensor v_j tem raio de transmissão menor que d(v_j, v_i)[1] então
            sensor v_j aumenta seu raio de transmissão para d(v_j, v_i)
```

4.6.2 EETA (*Energy-Efficient Test Assignment*)

Embora a estratégia TAWR apresente uma abordagem distribuída onde os sensores são capazes de gerar o assinalamento de testes com o mínimo de participação do *sink*, um número elevado de sensores, e, consequentemente, de testes, é utilizado. A forma de

[1] $d(v_j, v_i)$ é a distância euclidiana entre os sensores v_j e v_i.

delimitação da região R, por ser retangular, agrega sensores distantes que participam do processo de diagnóstico sem necessidade, gerando assim um gasto de energia que pode ser evitado.

Para realizar uma redução no custo energético utilizado pelo processo de diagnóstico, duas questões tem impacto direto: número de sensores em V_D e a proximidade geográfica destes sensores. Com um número menor de sensores em V_D, menos testes serão realizados. E, visto que a energia gasta em um teste é diretamente proporcional à distância geográfica entre os sensores testado e testador, quanto mais próximos os sensores de V_D estiverem entre si, menor o custo energético necessário.

A estratégia EETA (*Energy-Efficient Test Assignment)*, apresentada a seguir, visa a redução do custo energético necessário para a execução dos testes. Essa redução é alcançada pela escolha de um número limitado de sensores que participam do processo de testes. A escolha dos sensores de V_D considera também a posição geográfica dos mesmos buscando uma diminuição da distância entre sensores testados e testadores, possibilitando testes com um menor custo energético.

Para a definição da estratégia de testes, vamos assumir novamente que um conjunto T, de cardinalidade t, de sensores gera um alarme que é recebido pelo *sink*. Nesta estratégia, diferentemente da estratégia TAWR, o *sink* é responsável pela definição do assinalamento de testes. A execução da estratégia de testes apenas pelo *sink* permite uma economia de energia nos sensores da rede. Após a definição do conjunto de testes, o *sink* repassa aos sensores de V_D quais testes cada sensor deve executar.

O algoritmo de definição do grafo de testes D avalia o custo total de energia necessário pelos sensores para a execução dos testes. Ao receber o alarme gerado pelos sensores em T, o *sink* inicia o processo de formação do grafo de testes D. Primeiramente, é identificada a região geográfica R onde o alarme foi gerado. Esta região é definida como a menor área retangular que compreende todos os sensores de T. A Figura 4.4 mostra um exemplo de definição da região R, com $t = 5$. Na Figura, os círculos representam sensores (identificados por números). Sensores marcados com um "X" pertencem a T.

O *sink* utiliza a região R como uma forma de centralizar a escolha dos sensores que

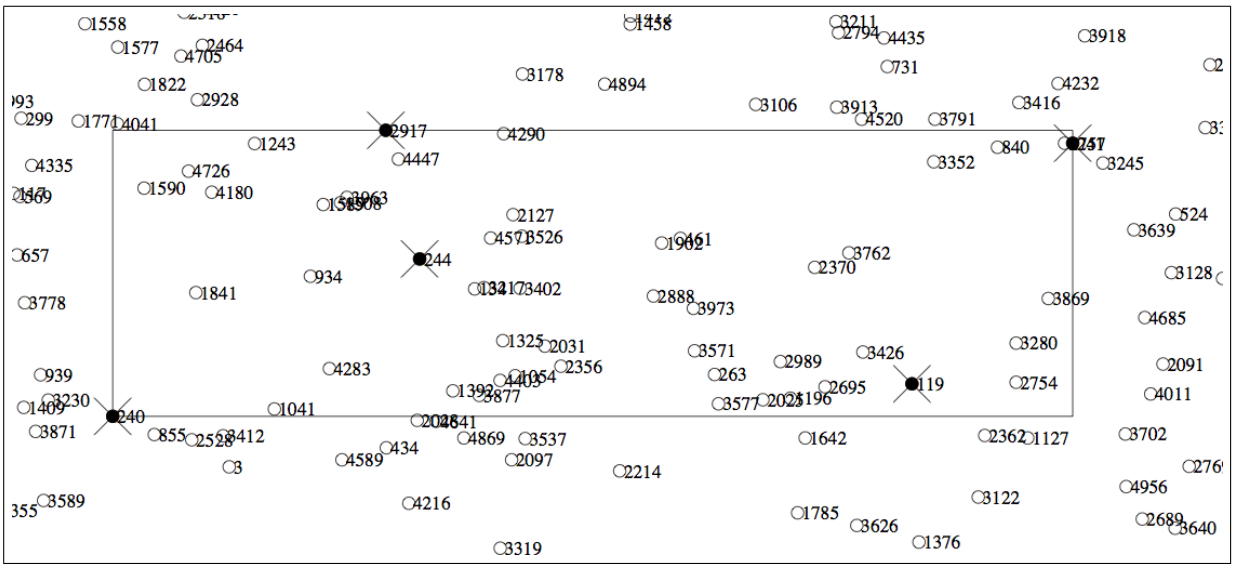


Figura 4.4: Exemplo de definição da região R, para $t = 5$.

participarão dos testes, ou seja, dos sensores de V_D. A escolha dos sensores de V_D utiliza como base o centro da região R, R_c. Através do uso de R_c a escolha dos sensores pode ser balanceada, permitindo que nenhum sensor de T fique demasiadamente distante dos demais sensores de V_D.

Os sensores de V são divididos em 4 grupos, ou quadrantes, utilizando como base da divisão o ponto R_c. A Figura 4.5 mostra um exemplo dessa divisão da rede em quadrantes a partir de R_c. Definimos V_i como o conjunto de sensores presentes no quadrante i, $(i = 0, \ldots, 3)$. Desta forma temos $V = V_0 \cup V_1 \cup V_2 \cup V_3$.

Novamente, o objetivo da divisão da rede em 4 quadrantes é permitir que sensores não executem testes recíprocos. Para isso, os testes são executados seguindo uma ordem entre os quadrantes, semelhante à estratégia TAWR. Cada quadrante possui 2 quadrantes vizinhos que, seguindo o sentido anti-horário, são nomeados como predecessor e sucessor. Um sensor presente em um quadrante irá executar testes apenas em sensores situados no seu quadrante sucessor, e será testado apenas por sensores posicionados em seu quadrante antecessor. Esta ordem garante que testes recíprocos não ocorram.

Para que o grafo de testes D seja t-diagnosticável, as condições $(c1)$ $(n \geq 2t+1)$ e $(c2)$ (o grau de D deve ser pelo menos t, ou seja, cada sensor deve ser testado por pelo menos t outros sensores) devem ser satisfeitas. Seguindo a divisão da rede em 4 quadrantes, é

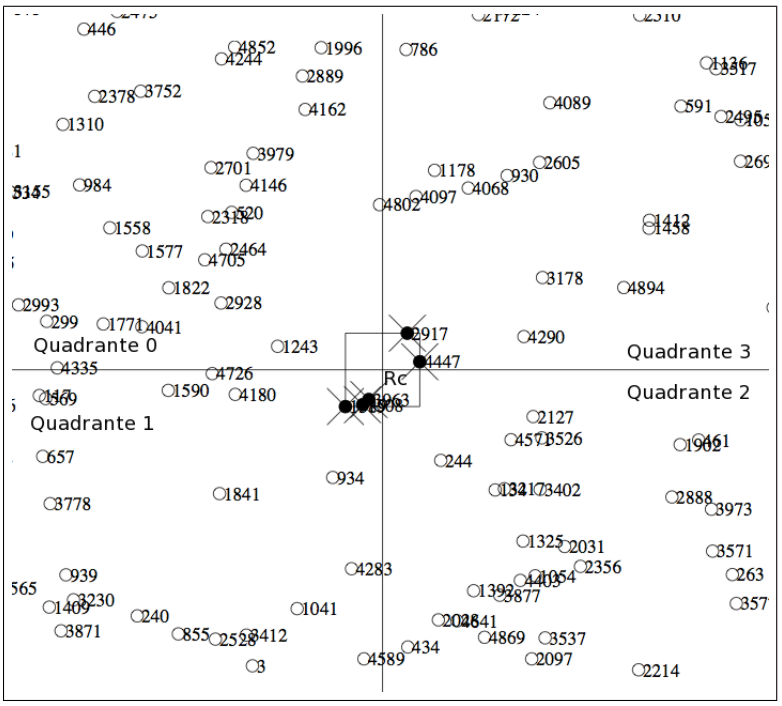


Figura 4.5: Exemplo da divisão da rede em quadrantes a partir de R_c.

necessário garantir que cada um dos quadrantes seja composto por pelo menos t sensores, satisfazendo assim ambas as condições. Com t sensores em cada quadrante, cada sensor pode ser testado por t outros ($c2$) e o número de sensores que participarão do diagnóstico é suficiente para satisfazer ($c1$), pois temos $4t \geq 2t + 1$. Uma vez que a estratégia visa diminuir o número de sensores de V_D, sempre apenas t sensores são selecionados em cada quadrante (diferentemente da estratégia TAWR, onde cada quadrante é passível de ter mais do que t sensores). Dessa forma temos $n = 4t$.

A estratégia visa, então, selecionar t sensores de cada V_i, para $i = 0, \ldots, 3$. Definimos como Q_i o conjunto de sensores posicionados no quadrante i selecionados para participar do processo de diagnóstico. Desta forma temos $V_D = Q_0 \cup Q_1 \cup Q_2 \cup Q_3$, com $Q_i \subset V_i$.

Para esta estratégia, o custo energético total gasto pelos testes realizados pelos sensores presentes no quadrante i é definido como C_{Q_i}, ou seja:

$$C_{Q_i} = \sum C_{i,j} \forall (v_i, v_j) \in E_D \mid v_i \in Q_i \ e \ v_j \in Q_{(i+1) \bmod 4} \tag{4.7}$$

onde $\bmod n$ é a operação módulo n. Definimos também o custo energético total utilizado por todos os testes presentes em D como:

$$C_T(D) = \sum_{i=0}^{3} C_{Q_i} \tag{4.8}$$

A seleção dos sensores de V_D é realizada de modo a obter uma diminuição do gasto energético total utilizado pelos testes entre os sensores.

Para que o diagnóstico seja possível, a condição $(c2)$ deve ser satisfeita. Sabendo que $|Q_i| = t$ pode-se concluir que um sensor $v_j \in Q_i$ irá realizar testes sobre todos os sensores presentes em $Q_{(i+1) \bmod 4}$, ou seja, $E_D = \{(v_j, v_k)\ \forall\ v_j \in Q_i$ e $\forall v_k \in Q_{(i+1) \bmod 4}$ para $i = 0, \ldots, 3\}$.

Deve-se observar que uma aresta $(v_i, v_j) \in E_D$ pode não existir inicialmente em E, ou seja, o sensor v_i pode não estar no raio de transmissão do sensor v_j e vice e versa. Para possibilitar que o sensor v_i teste v_j é necessário que ocorra um aumento nos raios de transmissão. Para testes bi-direcionais é necessário que ambos os sensores aumentem seus raios de transmissão para um valor suficiente. Já para testes uni-direcionais, apenas o raio de transmissão do sensor testado, v_j, necessita alteração. Desta forma, o *sink* solicita que os sensores que não são vizinhos entre si, mas que formam uma aresta em E_D, aumentem seus raios de transmissão para que o teste seja possível. Nesta estratégia não é necessário um algoritmo de aumento de raio de transmissão como o utilizado na estratégia TAWR, pois na estratégia EETA o *sink* é capaz de informar para cada sensor, especificamente, qual deve ser o raio de transmissão necessário para que todos os testes sejam executados com êxito.

Para que a seleção dos sensores em cada quadrante permita reduções em $C_T(D)$, uma heurística inicial é seguida. Primeiramente, os sensores de T são selecionados para os quadrantes em que estão posicionados, visto que tais sensores devem participar do processo de testes. Definimos, assim, T_i como sendo o conjunto de sensores de T posicionados geograficamente no quadrante i. A cardinalidade de T_i é igual a t_i. Assim temos: $T_i \subset Q_i$. Considerando o modelo energético apresentado neste trabalho, temos que o custo energético necessário para a execução de um teste entre dois sensores é proporcional à distância geográfica entre os sensores. Logo, selecionando sensores próximos geograficamente para V_D temos uma maior probabilidade de gerar um grafo de testes D com um

valor $C_T(D)$ reduzido.

A seleção inicial dos sensores de V_D é, então, baseada na distância dos sensores em relação a um ponto em comum, no caso R_c. A partir deste princípio, Q_i é formado inicialmente pelos t_i sensores de T_i e os $t - t_i$ sensores do quadrante i mais próximos geograficamente de R_c. Assim V_D será formado em sua maioria por sensores próximos ao centro R_c, aumentando a probabilidade de uma redução em $C_T(D)$. A Figura 4.6 mostra um exemplo de definição do conjunto V_D. Círculos escuros representam sensores pertencentes a V_D e números são os identificadores de cada sensor.

Embora a heurística inicial tenha como princípio selecionar os sensores mais próximos ao ponto central R_c, deve-se notar que um sensor $v_j \in T_i$ pode estar mais afastado de R_c do que um sensor $v_k \in (V_i - Q_i)$. Todo sensor $v_k \in T$ deve participar do processo de diagnóstico, não importando sua distância ao ponto R_c. A Figura 4.7 mostra um exemplo para este caso. Na Figura círculos pretos identificam sensores de V_D e círculos brancos representam sensores não selecionados para o diagnóstico. Observa-se na Figura que todos os sensores de $v_j \in T_i$ se encontram mais afastados de R_c do que alguns sensores $v_k \in (V_i - Q_i)$.

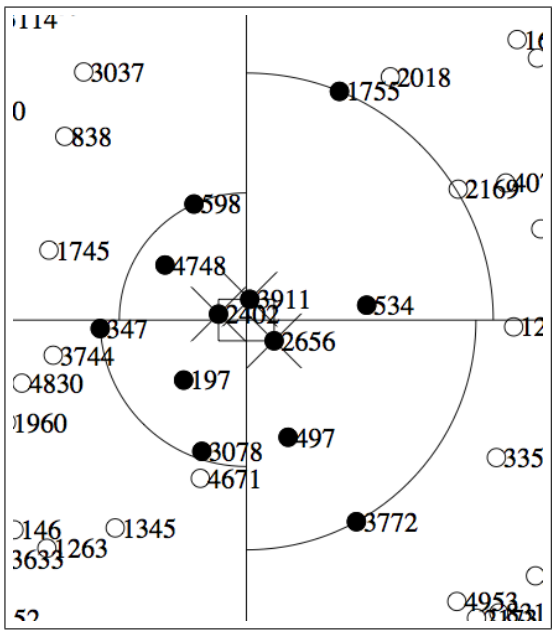


Figura 4.6: Definição do conjunto V_D inicial, formado pelos sensores de T (marcados com "X") e pelos sensores mais próximos geograficamente de R_c, para $t = 3$.

Apesar dos sensores selecionados pela heurística inicial serem os mais próximos do

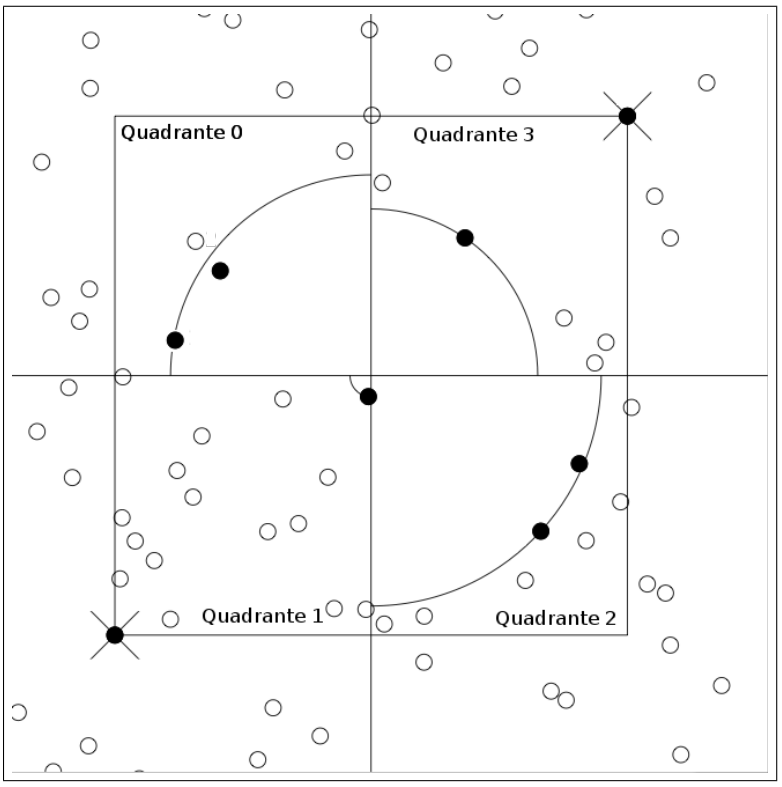


Figura 4.7: Exemplo de sensores de T afastados de R_c, para $t = 2$.

centro de R, sensores mais distantes podem estar a uma distância menor da maioria dos sensores presentes nos seus quadrantes antecessor e sucessor, podendo assim produzir custos menores de testes. Este caso pode ser visto na Figura 4.8. No exemplo da Figura, o sensor 2 do quadrante 3 está mais distante de R_c do que o sensor 1, porém está mais próximo da maioria dos sensores escolhidos do quadrante 0 e do quadrante 2. Logo, a troca do sensor 1 pelo sensor 2 pode gerar uma diminuição de $C_T(D)$. Para avaliar possíveis conjuntos de sensores que garantam um menor valor de $C_T(D)$ do que o obtido pela heurística inicial, um algoritmo é executado.

Considere os sensores $v_j \in V_i$ e $v_k \in V_i$. O sensor v_j é definido como sendo o sensor mais distante de R_c, porém pertencente a Q_i, e v_k é definido como sendo o sensor mais próximo geograficamente do sensor v_j, v_k é mais distante de R_c do que v_j, tal que $v_k \notin Q_i$. Em outras palavras, v_k é o sensor mais próximo de R_c que não foi selecionado para Q_i. O algoritmo consiste em verificar se a troca de algum sensor $v_n \in Q_i$ pelo sensor v_k produz redução em $C_T(D)$. A troca que produzir a maior redução em $C_T(D)$ é realizada. Se alguma troca for realizada, o processo se repete em busca de novas possíveis trocas no

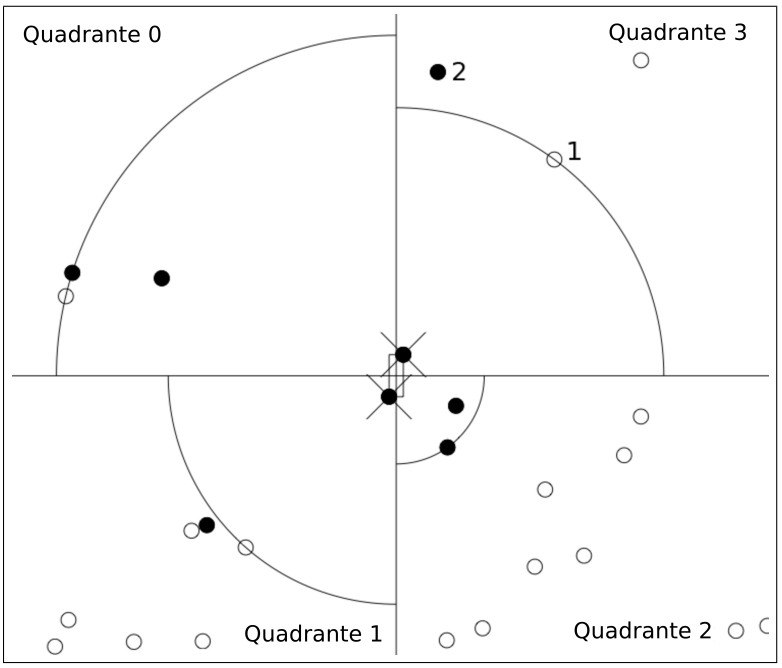


Figura 4.8: Exemplo de escolha de sensor mais distante de R_c, porém com um custo energético menor.

mesmo quadrante. Caso nenhuma troca seja realizada no quadrante i, o quadrante $(i + 1)$ mod 4, passa a ser analisado. O processo é encerrado uma vez que a troca do sensor $v_k \in V_i$, e $V_k \notin Q_i$, por algum sensor $v_n \in Q_i$, para $i = 0, \ldots, 3$, não gera redução em $C_T(D)$. O Algoritmo 2 apresenta o pseudo-código da estratégia EETA. Embora os quadrantes possam ser analisados mais de uma vez à procura de possíveis trocas, a estratégia é finita pois, em algum momento, qualquer sensor mais afastado de R_c produzirá custos maiores.

Apesar da estratégia de testes apresentada não garantir que o valor de $C_T(D)$ seja mínimo, ela proporciona uma redução considerável no custo energético quando comparado à estratégia TAWR. A redução no número de sensores que participam dos testes, a geração do grafo de testes pelo *sink* e o algoritmo de trocas de sensores garantem reduções no gasto de energia para a maioria dos casos. O Capítulo 5 apresenta resultados de simulações que comprovam tais reduções no custo energético.

Algoritmo 2: Estratégia EETA

```
R = menor região que compreende T
R_c = centro de R
para cada v em T faça
    i = quadrante de v
    Q_i = Q_i + {v}
para i = 0 até i = 3 faça
    enquanto |Q_i| < t faça
        v = sensor do quadrante i mais próximo de R_c | v ∉ Q_i
        Q_i = Q_i + {v}
reduz_custo = true
enquanto reduz_custo faça
    reduz_custo = false
    para i = 0 até i = 3 faça
        reduz_custo_quadrante = true
        enquanto reduz_custo_quadrante faça
            reduz_custo_quadrante = false
            custo = C_T(D)
            v = sensor mais próximo de R_c no quadrante i que nunca foi selecionado
            para Q_i
            para cada u ∈ Q_i faça
                Q_i = Q_i − {u}
                Q_i = Q_i + {v}
                novo_custo = C_T(D)
                se novo_custo < custo então
                    melhor_troca = u
                    custo = novo_custo
                    reduz_custo_quadrante = true
                Q_i = Q_i + {u}
                Q_i = Q_i − {v}
            se reduz_custo_quadrante então
                reduz_custo = true
                Q_i = Q_i − {melhor_troca}
                Q_i = Q_i + {v}
```

4.6.3 ODTA (*Optimal Design Test Assignment*)

Embora a estratégia EETA garanta uma diminuição no número de sensores utilizados no diagnóstico, comparado com a estratégia TAWR, a divisão da rede em quadrantes ainda exige o uso de um número maior de sensores do que o mínimo possível para o diagnóstico correto, $2t + 1$. A estratégia apresentada a seguir, chamada ODTA (*Optimal Design Test*

Assignment) visa reduzir o número de sensores que participam do diagnóstico ao mínimo possível, ou seja, o grafo de testes formado pela estratégia ODTA utiliza sempre $2t + 1$ sensores.

Para a definição da estratégia ODTA, vamos assumir novamente que um conjunto T, de cardinalidade t, de sensores gera um alarme que é recebido pelo *sink*. Igualmente à estratégia EETA, na estratégia ODTA o *sink* é responsável pela definição do grafo de testes D e irá informar a cada sensor selecionado para o processo de diagnóstico quais testes devem ser executados.

Para realizar a definição do algoritmo de formação do grafo de testes D da estratégia ODTA, utilizamos os conceitos de sistemas ótimos e de *designs* ótimos [8]. No contexto de diagnóstico em nível de sistema, um sistema S, composto por n unidades, é definido como ótimo se $n = 2t + 1$, onde t é o número de unidades falhas existentes em S, e cada unidade de S é testada por exatamente t outras unidades. Um sistema ótimo é definido por um *design* ótimo, ou seja, um conjunto de arestas, ou testes, que torna S ótimo. Em geral existem vários *designs* ótimos para um mesmo sistema S. Assim, Preparata et al. [8] definem $D_{\delta t}$ como um conjunto, ou família, de *designs* ótimos. Um sistema S pertence a um *design*, ou grafo, $D_{\delta t}$ quando um teste (u_i, u_j) existe em S se e somente se $(j - i) \bmod n = (\delta m) \bmod n$ com m assumindo os valores $1, 2, \ldots, t$.

Preparata et al. provam que, se um sistema S emprega um grafo D_{1t}, então S é t-diagnosticável [8]. A prova é descrita a seguir. Grafos D_{1t} apresentam um aspecto cíclico de testes. Seguindo a regra de formação de *designs* descrita acima, em um grafo D_{1t} composto por n unidades, $u_0, u_1, \ldots, u_{n-1}$, uma unidade qualquer u_i irá realizar testes nas suas t próximas unidades, (seguindo a ordem de seus identificadores) e será testada por suas t unidades anteriores (a Figura 4.9 apresenta um exemplo de um assinalamento de testes gerado por um grafo D_{1t} para um sistema com 5 unidades). Assim, a seguinte sequência cíclica de testes irá existir: $u_0 \rightarrow u_1 \rightarrow, \ldots, \rightarrow u_{n-1} \rightarrow u_0$, onde $u_0 \rightarrow u_1$, indica que a unidade u_0 testa a unidade u_1. Considere agora, com relação à sequência de testes apresentada, a existência de uma sequencia de b unidades falhas, precedida e sucedida por unidades sem-falha, com $b < t$. Sendo $u_i, u_{i+1}, \ldots, u_{i+b-1}$ as unidades falhas, logo,

temos que u_{i-1} e u_{i+b} são unidades sem-falha. Seguindo a regra de formação de grafos D_{1t}, temos que: $(i+b)-(i-1) = b+1 \leq t$, logo, existe uma aresta no grafo de testes ligando as unidades sem-falha u_{i-1} e u_{i+b}. Desta forma, conclui-se que sempre, em um grafo D_{1t}, será possível formar uma sequência de s testes, onde $s \geq t+1$, entre unidades sem-falha. Considerando que tal sequência de testes indica que $t+1$ unidades sem-falha foram encontradas, pode-se assumir que todos os testes desta sequência são confiáveis, pois, por hipótese, não existem mais do que t unidades falhas. Assim, cada unidade falha será testada por, no mínimo, uma unidade sem-falha, concluindo o diagnóstico.

É considerada, agora, a existência de uma sequência de t unidades falhas, digamos $u_i, u_{i+1}, \ldots, u_{i+t-1}$. Isto implica que as demais unidades, $u_{i+t}, u_{i+t+1}, \ldots, u_{i-1}$ (todos os cálculos dos identificadores são executados sobre módulo de n), são sem-falha. Assim, o grafo de testes irá reportar uma sequência de testes que indica que $t+1$ unidades sem-falha realizam testes entre si sequencialmente. Suponha agora que u_{i-1} também é falho. Desta forma, todos os testes que indicam que u_{i-1} é sem-falha estão incorretos, ou seja, todas as demais unidades $u_{i+t}, u_{i+t+1}, \ldots, u_{i-2}$ também são unidades falhas, pois apresentaram testes não confiáveis. Logo, teríamos mais do que t unidades falhas, o que contradiz a hipótese de que o sistema tem no máximo t unidades falhas. Assim, conclui-se que o sistema que emprega grafos D_{1t} é t-diagnosticável.

Apesar da prova apresentada não tratar da existência de testes recíprocos, pode-se concluir que, através de seu aspecto cíclico, grafos do tipo D_{1t} produzem um grafo de testes t-diagnosticável e sem testes recíprocos. Em seu trabalho, Preparata et al. provam, ainda, que grafos $D_{\delta t}$ também geram assinalamentos t-diagnosticáveis se δ e t são primos relativos [8]. Para essa prova, demonstra-se que os grafos D_{1t} são isomorfos aos grafos $D_{\delta t}$ quando δ e t são primos relativos.

Considerando que o custo energético total $C_T(D)$ tende a crescer proporcionalmente ao número de sensores que participam do diagnóstico, visto que mais testes são necessários, nota-se que o algoritmo de definição de D deve minimizar o número de sensores presentes em V_D. Para tanto, a estratégia ODTA faz uso, principalmente, de *designs* ótimos propostos em [8], os quais permitem participação de um número mínimo de unidades para

o processo de diagnóstico. Mais precisamente, a estratégia ODTA tem como objetivo definir o conjunto de arestas que configura um grafo D_{1t} para um sistema composto pelas unidades de V_D.

Supondo que o conjunto V_D já esteja definido e que $n = 2t+1$, para que a definição de um grafo de testes D pertencente a um grafo D_{1t} seja possível, cada sensor presente em V_D deve receber um identificador numérico único i, onde $i = 0 \ldots 2t$. Este identificador pode ser atribuído de forma aleatória pelo *sink*, apenas garantindo que não existam identificadores duplicados. A partir desses identificadores, uma rede overlay pode ser definida sobre os sensores de V_D verificando quais arestas devem existir no grafo de testes seguindo a definição de D_{1t} dada anteriormente.

O processo de formação do grafo D, segundo o grafo D_{1t} é executado da seguinte forma: cada sensor irá testar os t próximos sensores seguindo a ordem crescente de seus identificadores. Desta forma, um sensor v_i, onde i é seu identificador na rede overlay, irá testar os seguintes sensores: $v_{(i+1) \bmod n}, \ldots, v_{(i+t) \bmod n}$ da rede overlay. Desta forma, cada sensor irá testar t sensores e será testado por outros t sensores. O custo energético total, para a estratégia ODTA, é definido como:

$$C_T(D) = \sum C_{i,j} | \forall (v_i, v_j) \in E_D \tag{4.9}$$

A não existência de testes recíprocos é proporcionada pelo fato de que um sensor v_i testa seus t próximos sensores e é testado por seus t sensores anteriores na rede overlay. O mesmo ocorre para todos os sensores, que, de forma cíclica, evitam testes recíprocos. A Figura 4.9 mostra um exemplo do grafo de testes representado sobre a rede overlay, enquanto a Figura 4.10 apresenta um exemplo do mesmo grafo de testes sobre os sensores de uma rede. Nota-se que os grafos das duas Figura são isomorfos entre si.

Embora a atribuição dos identificadores não tenha nenhuma relação com a posição geográfica de cada sensor de V_D, a estratégia ODTA garante que, dado um conjunto V_D, o assinalamento de testes de menor custo energético total para o conjunto V_D é gerado.

Teorema 1 *Dado um conjunto de sensores V_D de cardinalidade igual a $2t+1$, a estratégia*

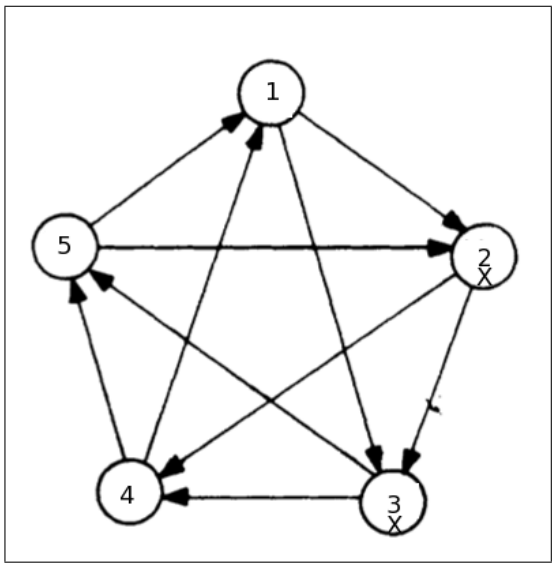


Figura 4.9: Grafo de testes gerado pela estratégia ODTA, representado sobre a rede overlay, para $n = 5$ e $t = 2$.

ODTA gera um grafo de testes D cujo $C_T(D)$ é mínimo para V_D.

Prova: Considerando que o algoritmo apresentado cria um assinalamento de testes com *design* ótimo, temos que cada sensor de V_D irá testar t sensores e será testado por t outros, evitando, assim, o uso de testes recíprocos. Logo, o grau de cada vértice presente em D é igual a $2t$. Assim, um sensor $v_i \in V_D$ terá uma relação de testado ou de testador com todos os demais sensores de V_D, o que garante que o grafo D, ignorando o sentido de suas arestas, seja sempre um grafo completo. Considerando que $C_{i,j} = C_{j,i}$, conclui-se que não existe um grafo de testes D com um conjunto diferente de arestas do gerado pelo algoritmo apresentado, que satisfaça as condições $(c1)$ e $(c2)$. Logo, não existe um conjunto de arestas para D que garanta um valor de $C_T(D)$ menor do que o obtido pela estratégia ODTA. ■

Dado que o algoritmo apresentado gera o assinalamento de testes de menor custo total para V_D, a escolha dos sensores que formam o conjunto V_D é a operação responsável pelo limite inferior do consumo total de energia utilizado no processo de diagnóstico.

O alto número de combinações possíveis para a escolha dos $2t + 1$ sensores que irão participar do processo de diagnóstico de forma a garantir que $C_T(D)$ seja mínimo sugere que o problema seja de ordem NP-Completo. Esta prova não será apresentada aqui, sendo considerada um trabalho futuro. Assim, uma heurística para a escolha de tais sensores é

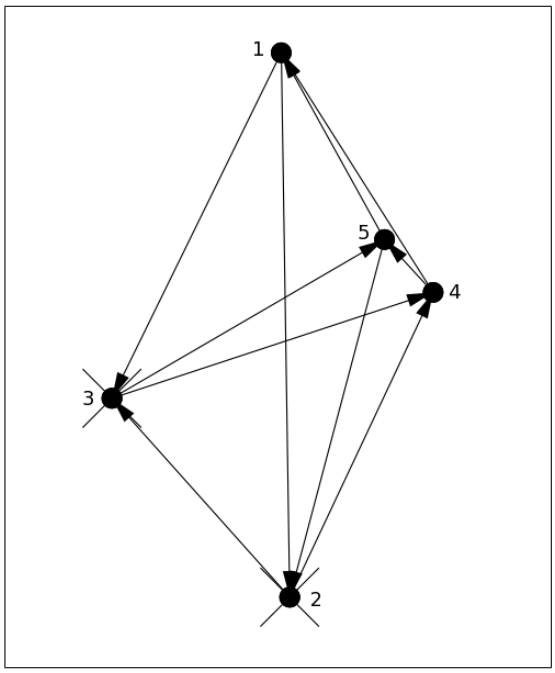


Figura 4.10: Grafo de testes gerado pela estratégia ODTA, representado sobre a rede de sensores, para $n = 5$ e $t = 2$.

utilizada. A heurística aqui apresentada proporciona a escolha de um conjunto de sensores onde a maioria deles está próxima entre si.

Tal heurística baseia-se em definir a menor região retangular que compreende todos os sensores de T. O centro de R, R_c, é utilizado como parâmetro para a escolha dos sensores. Como todos os sensores de T devem participar do processo de diagnóstico, a heurística é responsável pela escolha dos $t + 1$ sensores restantes. Assim, são escolhidos os $t + 1$ sensores mais próximos geograficamente do ponto R_c e que não estejam presentes em T.

Como o conjunto V_D, ao final do processo de escolha dos sensores, terá $2t+1$ sensores, conclui-se que a heurística garante que, pelo menos, a maioria dos sensores $(t + 1)$ são os mais próximos geograficamente entre si. Logo, o custo da maioria das arestas, ou testes, de D também será minimizado. Porém, não se pode afirmar que o custo total alcançado seja mínimo.

Duas situações específicas podem ser nomeadas como pior e melhor caso para a heurística apresentada. A primeira (pior caso) é definida quando os sensores de T estão distantes entre si. Já a segunda (melhor caso) compreende as ocasiões em que sensores de T estão todos próximos entre si.

Para o melhor caso, a heurística irá apresentar um conjunto V_D de sensores que, com alta probabilidade, possui o menor valor de $C_T(D)$ para realizar o diagnóstico. Isto é possível porque os sensores mais próximos de R_c serão também os mais próximos dos sensores de T, minimizando, assim, o custo de praticamente todas as arestas de D.

Para o pior caso, a heurística, como dito anteriormente, irá garantir que, pelo menos, a maioria dos sensores de V_D sejam os mais próximos entre si. Porém, considerando aplicações reais e sensores sem-falha, é pouco provável que os sensores de T estejam distantes entre si, visto que se um fenômeno monitorado pelos sensores realmente ocorrer em uma região da rede, é provável que todos os sensores desta região detectem o fenômeno. Logo, o pior caso é pouco provável em aplicações reais.

CAPÍTULO 5

EXPERIMENTOS E DISCUSSÃO

No Capítulo 4 foram apresentadas 3 diferentes abordagens capazes de definir um assinalamento de testes t-diagnosticável, com um consumo minimizado de energia. Neste Capítulo serão descritos experimentos e seus resultados, que avaliam e comparam o custo energético e outras propriedades das estratégias apresentadas.

Este Capítulo está organizado da seguinte maneira: a Seção 5.1 descreve o ambiente de simulação utilizado. Os resultados dos experimentos são apresentados na Seção 5.2. A Seção 5.3 apresenta uma discussão sobre os resultados apresentados.

5.1 Ambiente de Simulação

Os resultados apresentados neste Capítulo foram obtidos através de simulações. O simulador utilizado, programado em linguagem C++, tem como função gerar posicionamentos geográficos aleatórios para sensores de uma rede. Em seguida, o simulador seleciona t sensores, também de forma aleatória, para formar o conjunto T. A partir dos t sensores selecionados e da posição geográfica dos demais sensores da rede, o simulador implementa as abordagens de testes apresentadas. Para cada estratégia, o simulador é capaz de estimar, entre outras propriedades, o gasto energético de cada sensor envolvido no processo de diagnóstico.

Para a geração das redes, o simulador recebe um conjunto de parâmetros. Entre eles estão: (1) tamanho da rede, que define o tamanho (em metros) do campo de sensoriamento em que os sensores serão depositados; (2) número de sensores; (3) t, ou seja, número de sensores que reportam mensagens de alarme; (4) distribuição estatística utilizada para posicionar geograficamente de forma aleatória os sensores no campo de sensoriamento; e (5) o *fator da região de alarme* (FRA), que define o tamanho da região da qual os sensores de T devem ser escolhidos.

Mais especificamente, a *região de alarme* é definida como uma região quadrada posicionada aleatoriamente sobre a rede. Apenas os sensores posicionados dentro da região de alarme são candidatos a serem escolhidos como sensores de T. O fator da região de alarme, ou FRA, é um número que indica a proporção, com relação ao tamanho da rede, do tamanho da região de alarme. Assim, com FRA igual a $0,1$, uma região quadrada com 10% do tamanho da rede será definida de forma aleatória para a seleção do sensores de T. Desta forma, o valor de FRA tem implicação direta na localidade espacial dos sensores de T.

Duas distribuições estatísticas são suportadas pelo simulador: distribuição uniforme e distribuição triangular [80]. A distribuição uniforme permite que a rede seja composta por sensores distribuídos de uma maneira homogênea sobre o campo de sensoriamento. Por sua vez, a distribuição triangular possibilita a geração de redes com uma concentração maior de sensores em algum ponto especifico permitindo, assim, a simulação de casos de aglomeração de sensores em pontos da rede. Para isso, a distribuição triangular recebe três parâmetros: um valor mínimo a, um valor máximo b e uma moda c, de modo que a função de densidade da distribuição é igual a zero para os extremos a e b, e obtém valor máximo em c, formando um triângulo em seu gráfico. O gráfico da função densidade de probabilidade da distribuição triangular é apresentada na Figura 5.1.

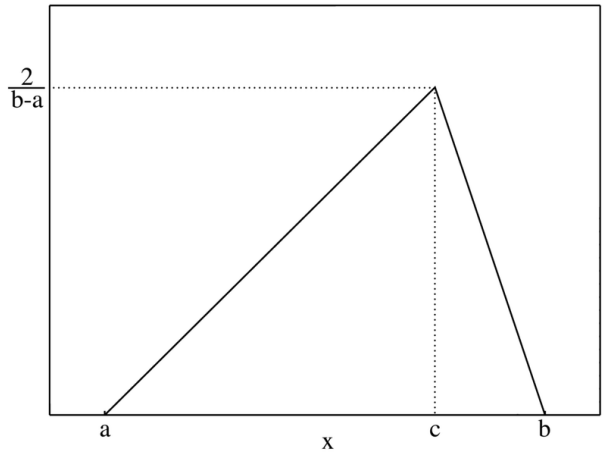


Figura 5.1: Função densidade de probabilidade da distribuição triangular.

5.1.1 Propriedades Avaliadas nas Simulações

Os experimentos executados visam avaliar e comparar as abordagens de testes apresentadas neste trabalho. As propriedades avaliadas em cada experimento, para diferentes valores de t, são:

- custo energético total do assinalamento de testes D ($C_T(D)$);
- custo energético médio por sensor do assinalamento de testes D ($C_M(D)$);
- número de sensores presentes no grafo de testes;
- impactos de diferentes valores de FRA.

A avaliação do custo energético total visa estabelecer uma análise da quantidade total de energia que é consumida por cada abordagem durante o processo de diagnóstico. Assim, é possível observar o consumo de energia global da rede, e estimar o impacto de tal custo no tempo de vida útil da mesma. A avaliação do custo energético médio entre os sensores, por sua vez, tem como objetivo verificar a distribuição do consumo energético entre os sensores. Por exemplo, é possível observar se, mesmo com um número grande de sensores participando do diagnóstico, existe uma média de consumo elevado por sensor. O estudo do número de sensores presentes no grafo de testes permite uma análise da escalabilidade da estratégia, visto que quanto mais sensores participam do processo, mais testes são necessários e, por conseguinte, uma comunicação mais intensa entre os sensores é realizada. Por fim, a avaliação do impacto de diferentes valores de FRA visa estudar o comportamento das abordagens e de seus consumos energéticos em ambientes com diferentes configurações de T.

Diferentes conjuntos de simulações foram executados, e para cada um, as propriedades 1 a 4 acima enumeradas foram avaliadas. Para cada caso de teste ou conjunto de parâmetros foram geradas 100 redes diferentes.

Em todas as simulações, redes de tamanho $512mX512m$ compostas por 512 e 1024 sensores foram consideradas. Simulações foram executados com os seguintes valores de t: 1, 3, 5, 8, 10, 12 e 15; e com os seguintes valores para FRA: 0,01, 0,1, 0,5 e 1.

Os valores coletados nas simulações da estratégia TAWR levam em consideração o melhor conjunto de testes possível para a situação, ou seja, o conjunto de testes de menor custo. Considerando o comportamento distribuído de TAWR, onde sensores solicitam testes, vários grafos de testes diferentes podem ser formados. Assim, foi utilizado para a comparação com as demais estratégias sempre o melhor caso, que é formado pelos testes de menor consumo energético e onde cada sensor é testado por, exatamente, t outros sensores. Em aplicações reais, os grafos de testes gerados pela estratégia TAWR podem apresentar consumos de energia maiores do que os apresentados nesses resultados. Também não foram considerados, para a estratégia TAWR, os custos de formação do grafo de testes (processo em que sensores solicitam testes). Foi avaliado apenas o consumo propiciado pelo conjunto final de testes do melhor caso. A Tabela 5.1 lista o conjunto de parâmetros das simulações.

Parâmetro	Valores
t	1,3,5,8,10,12,15
FRA	0,01, 0,1, 0,5, 1
Distribuição	Uniforme, Triangular
Dimensão do campo de sensoriamento	$512mX512m$
Número de sensores	512, 1024
Número de redes geradas por experimento	100

Tabela 5.1: Parâmetros das simulações.

5.2 Resultados

Esta Seção apresenta os resultados obtidos através das simulações e suas conclusões. Primeiramente, as Seções 5.2.1 e 5.2.2 demonstram resultados obtidos para redes geradas por distribuição uniforme e triangular, respectivamente. Por fim, a Seção 5.2.3 descreve o comportamento apresentado pelas abordagens de testes com diferentes valores de FRA.

5.2.1 Distribuição Uniforme

No conjunto de experimentos que é apresentado nesta Seção, a posição geográfica dos sensores foi definida através de distribuição uniforme. A Figura 5.2 mostra um exemplo

do posicionamento geográfico gerado pela distribuição uniforme em um dos experimentos. Na Figura, sensores são representados por pontos. Nota-se um posicionamento homogêneo dos sensores por todo o campo de sensoriamento, que possui $512mX512m$.

Para estes experimentos, o valor de FRA foi fixado em $0,1$. Desta forma, os casos simulados supõem que os sensores de T são escolhidos a partir de uma região quadrada de $51,2mX51,2m$, o que garante certa proximidade entre os mesmos. Esta asserção está baseada no fato de que, dependendo da aplicação, sensores sem-falha próximos entre si irão detectar um mesmo fenômeno. Simulações com 512 e 1024 sensores foram executadas.

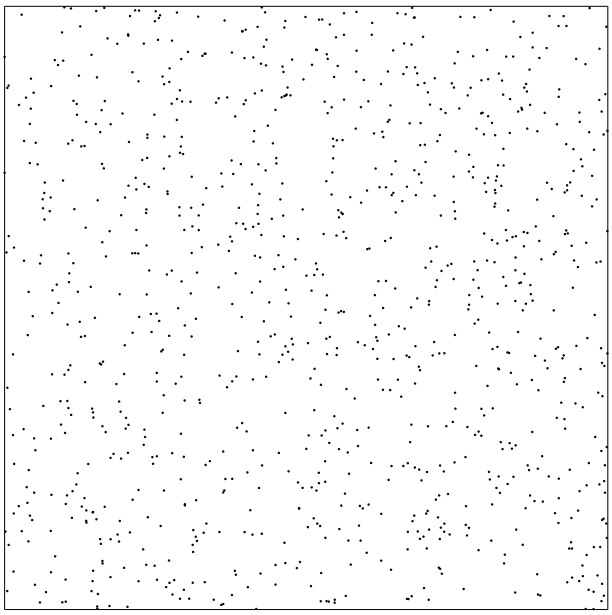


Figura 5.2: Exemplo de posicionamento de 1024 sensores gerado por distribuição uniforme.

5.2.1.1 Custo Energético Total

A Figura 5.3 apresenta a média do custo energético total ($C_T(D)$) obtido pelas três estratégias para redes com 512 e 1024 sensores e diferentes valores de t. Os resultados mostram que o consumo total de energia cresce proporcionalmente ao valor de t em todas as estratégias, tanto para 512 sensores quanto para 1024. Este resultado, de certa forma esperado, ocorre pelo fato de que o número de sensores e, consequentemente, de testes utilizados aumenta com valores maiores de t, gerando um crescimento no consumo de energia.

Nota-se que todas as estratégias apresentam uma redução do custo energético total em redes mais densas devido à maior proximidade entre os sensores. A estratégia TAWR apresenta um consumo maior que as demais estratégias em todos os casos. Isso ocorre pelo fato de o número de sensores utilizados pela estratégia TAWR ser maior do que o número apresentado pelas estratégias EETA e ODTA. Apesar da estratégia EETA apresentar uma redução no consumo de energia em comparação à TAWR, seu consumo ainda é consideravelmente maior do que o apresentado pela estratégia ODTA, que utiliza o número mínimo de sensores para o diagnóstico. A abordagem ODTA apresenta consumos totais de energia consideravelmente menores que as demais estratégias, inclusive, os valores obtidos pela estratégia ODTA para redes com 512 sensores são menores do que os valores alcançados pelas demais estratégias para redes com 1024 sensores.

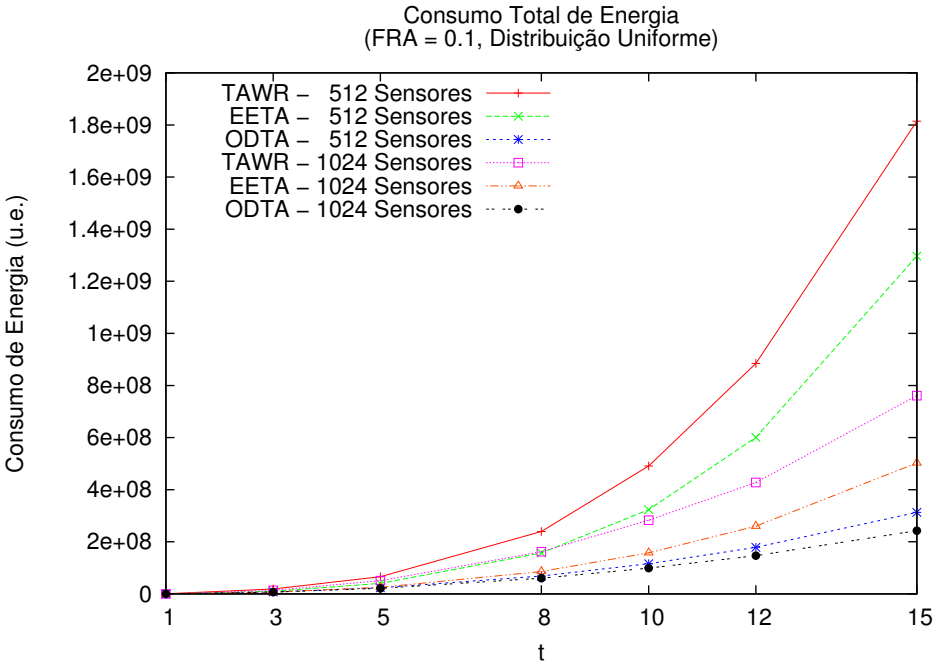


Figura 5.3: Consumo total de energia de cada estratégia, para redes de 512 e 1024 sensores e distribuição uniforme.

5.2.1.2 Número de Sensores Utilizados

A Figura 5.4 mostra o número médio de sensores utilizados pelas estratégias durante os experimentos. Fica claro o uso de um número maior de sensores pela estratégia TAWR em ambos os cenários. A definição da região retangular R e o comportamento distribuído

da estratégia TAWR causa o uso de um número maior de sensores. Para EETA e ODTA o número de sensores utilizados depende apenas do valor de t. Como visto no Capítulo 4, o número de sensores necessários é $4t$ sensores para EETA e $2t + 1$ sensores para ODTA.

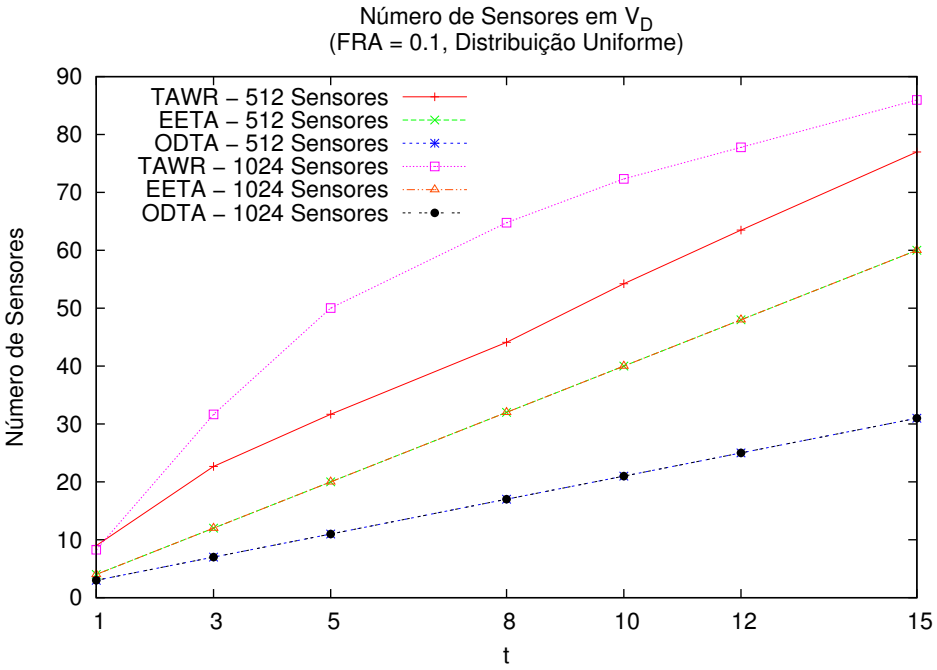


Figura 5.4: Número de sensores utilizados por cada estratégia, para redes de 512 e 1024 sensores e distribuição uniforme.

5.2.1.3 Custo Energético Médio por Sensor

Os resultados obtidos apontam, também, que o consumo médio $C_M(D)$ obtido entre os sensores é novamente maior para a estratégia TAWR. A Figura 5.5 mostra o custo energético médio entre os sensores ($C_M(D)$) obtido pelas 3 abordagens.

Para 512 sensores, TAWR e EETA apresentam valores mais altos devido ao menor número de sensores existentes na rede e, consequentemente, à maior distância entre os mesmos. Por outro lado, para redes mais densas, TAWR utiliza mais sensores e, apesar de obter um custo total maior, possibilita, em alguns casos, uma média de consumo menor que as demais estratégias.

A estratégia EETA não utiliza tantos sensores quanto TAWR, porém seu baixo custo total também permite valores menores de custo médio por sensor para redes mais densas. Para ODTA, o baixo número de sensores utilizados, o posicionamento mais distante entre

os sensores de T e a heurística de escolha de sensores faz com que o custo energético médio por sensor seja maior em alguns casos.

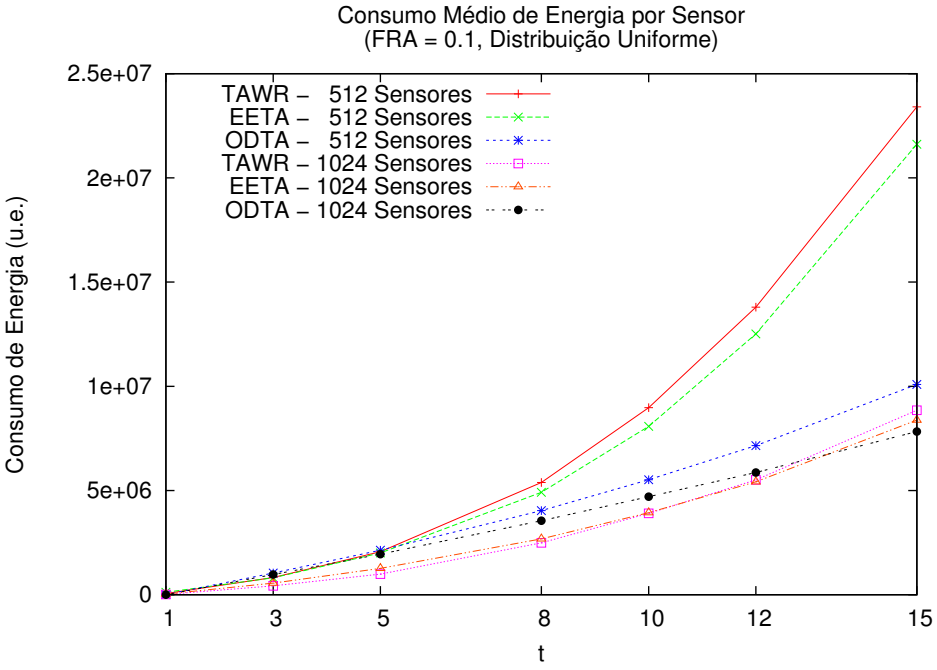


Figura 5.5: Consumo médio de energia de cada estratégia, para redes de 512 e 1024 sensores e distribuição uniforme.

5.2.2 Distribuição Triangular

Para o próximo conjunto de experimentos, o posicionamento dos sensores foi realizado através da distribuição triangular [80]. Os parâmetros utilizados pela distribuição foram selecionados de maneira a garantir que os sensores fossem aglomerados no centro do campo de sensoriamento. A Figura 5.6 mostra o posicionamento geográfico obtido em um dos experimentos através da distribuição triangular. Observa-se que o centro da rede é mais denso e que não existem sensores nas extremidades do campo de sensoriamento.

5.2.2.1 Custo Energético Total

A Figura 5.7 mostra a média do consumo total obtido pelas estratégias em ambientes formados pela distribuição triangular. Nestes experimentos a estratégia TAWR apresenta valores muito maiores que as demais estratégias. A alta concentração de sensores em um

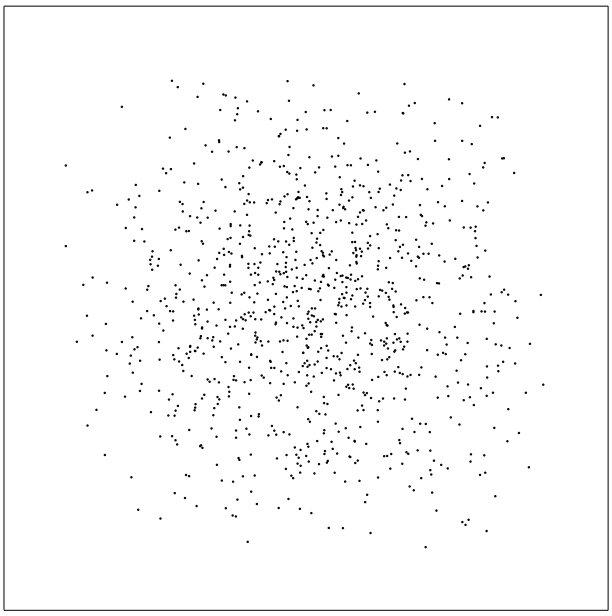


Figura 5.6: Exemplo de posicionamento de 1024 sensores gerado por distribuição triangular.

ponto da rede faz com que um número elevado de sensores sejam selecionados pela região R.

Especificamente para a estratégia TAWR, um comportamento diferenciado ocorre para situações com altas concentrações de sensores em determinados pontos da rede. Quando sensores de T estão situados mais próximos das extremidades da rede, a região R cresce demasiadamente em direção ao centro da rede até que todos os quadrantes tenham pelo menos t sensores. Assim, alguns quadrantes abrangem o centro da rede, selecionando um alto número sensores para V_D. A Figura 5.8 mostra a definição da região R pela estratégia TAWR em um desses casos. Na Figura, sensores marcados com "X" pertencem a T. É possível notar na Figura que um dos quadrantes contém um número elevado de sensores quando com parado aos demais quadrantes. A ocorrência deste fenômeno causa um aumento no gasto energético da estratégia TAWR.

Para as demais estratégias, que utilizam um número controlado de sensores, a alta concentração de sensores favorece a redução no consumo de energia. Tanto EETA quanto ODTA apresentam reduções no consumo total, comparados com resultados obtidos com distribuição uniforme, porém, para distribuição triangular, EETA continua apresentando valores de custo energético total maiores comparados com ODTA.

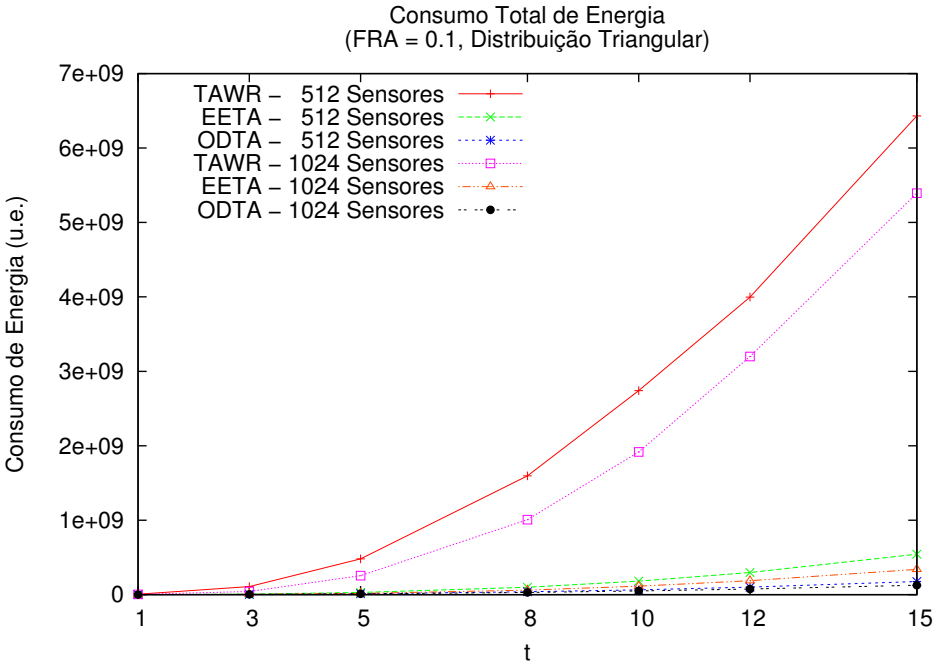


Figura 5.7: Consumo total de energia de cada estratégia, para redes de 512 e 1024 sensores e distribuição triangular

5.2.2.2 Número de Sensores Utilizados

A Figura 5.9 mostra o número médio de sensores utilizados pelas estratégias. Para TAWR, em redes com 1024 sensores, o número de sensores utilizados é maior para valores altos de t, pois, tanto o distanciamento dos sensores de T quanto seus posicionamentos nas extremidades da rede, geram um aumento de tamanho da região R, abrangendo mais sensores. Apesar disso, a proximidade dos sensores e o fato de que nem todos executam testes, permite um custo total menor do que em redes com 512 sensores. Novamente, o número de sensores utilizados por ODTA e EETA são fixos em relação a t.

5.2.2.3 Custo Energético Médio por Sensor

O custo energético médio por sensor ($C_M(D)$) obtido pelas estratégias é apresentado na Figura 5.10. Devido ao alto custo total, nem mesmo o uso de um número maior de sensores reduz o custo médio por sensor obtido pela estratégia TAWR, que, agora, apresenta os maiores valores. A alta densidade favorece a estratégia ODTA que utiliza poucos testes e sensores próximos entre si. A estratégia EETA apresenta resultados pouco maiores do

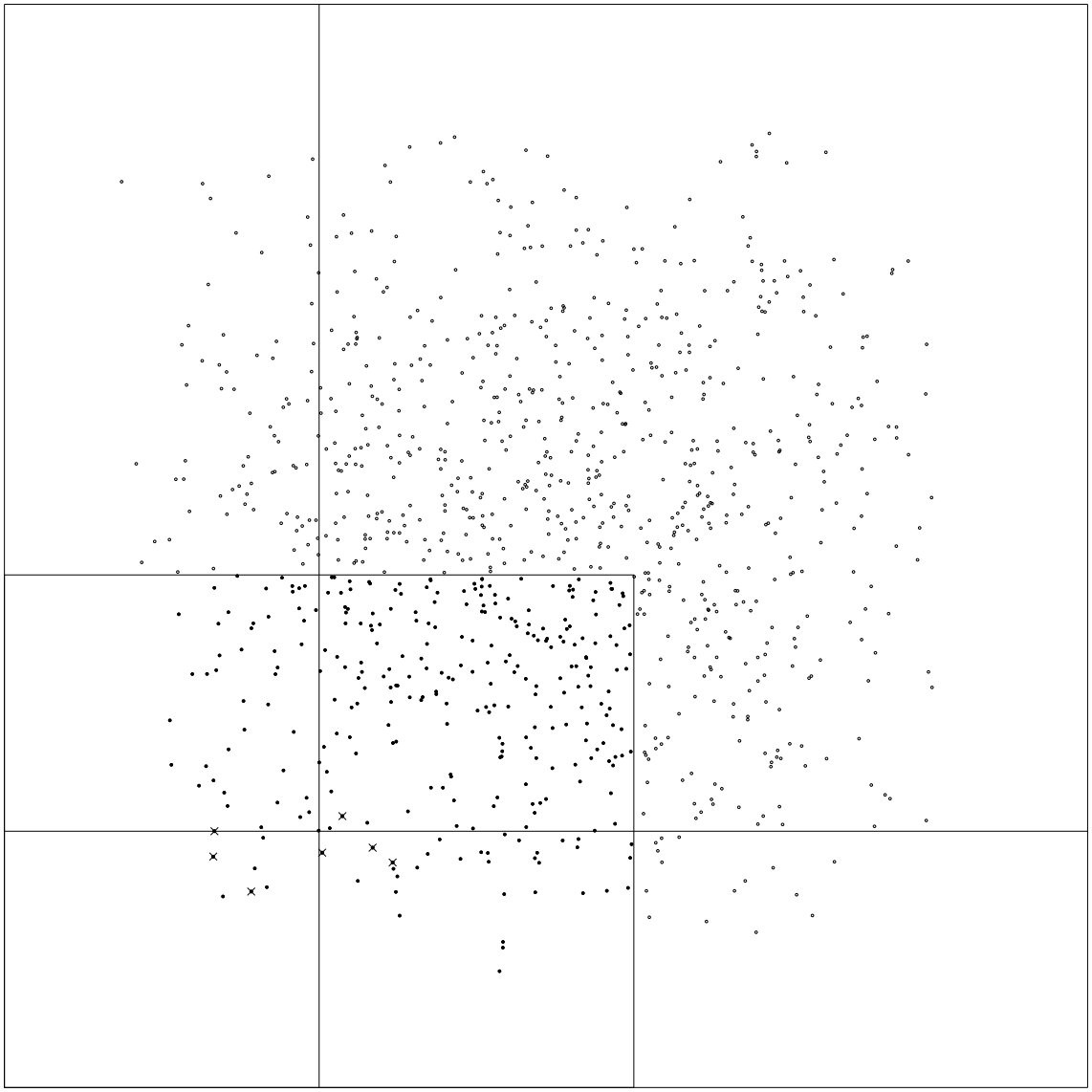


Figura 5.8: Exemplo aplicação da estratégia TAWR para distribuição triangular. Sensores de T localizados na extremidade da rede.

que ODTA.

5.2.3 Região de Alarme

Nesta Seção é apresentada uma comparação do comportamento das abordagens de testes para diferentes valores de FRA. O fator de região de alarme, define, de modo geral, a distância entre os sensores de T. Altos valores de FRA permitem que sensores de T estejam distantes entre si. Para o conjunto de experimentos mostrado a seguir, foram utilizadas simulações com 1024 sensores e distribuição uniforme.

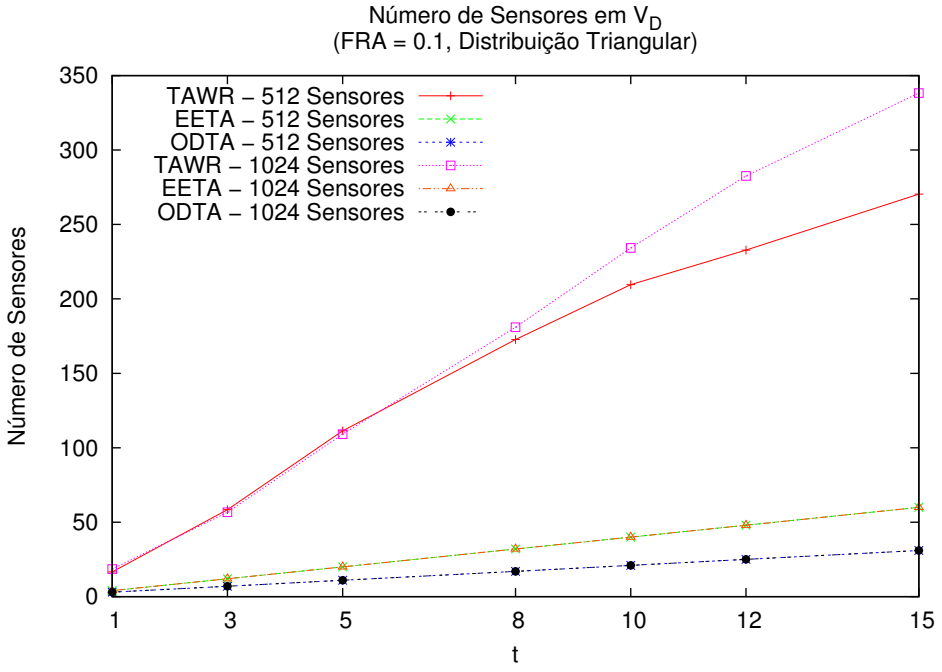


Figura 5.9: Número de sensores utilizados por cada estratégia, para redes de 512 e 1024 sensores e distribuição triangular

5.2.3.1 Custo Energético Total para Diferentes Valores de FRA

As Figuras 5.11, 5.12 e 5.13 mostram a variação do custo total obtido pelas estratégias TAWR, EETA e ODTA, respectivamente, para diferentes valores de FRA. Cada curva presente nos gráficos representa os resultados para um determinado valor de FRA.

Nota-se que, em todas as estratégias, o consumo total cresce consideravelmente com o aumento dos valores de FRA. A delimitação da região R baseada na posição dos sensores de T causa o uso de mais sensores em TAWR, e de testes mais "longos", e de maior custo, em EETA e ODTA.

5.2.3.2 Número de Sensores Utilizados por TAWR para Diferentes Valores de FRA

Na estratégia TAWR, o número de sensores utilizados cresce sensivelmente com valores altos de FRA. Pois, para abranger todos os sensores de T, distantes entre si, necessita-se que a região R seja demasiado grande, o que agrega um alto número de sensores para V_D.

A Figura 5.14 mostra o número de sensores utilizados pela estratégia TAWR com

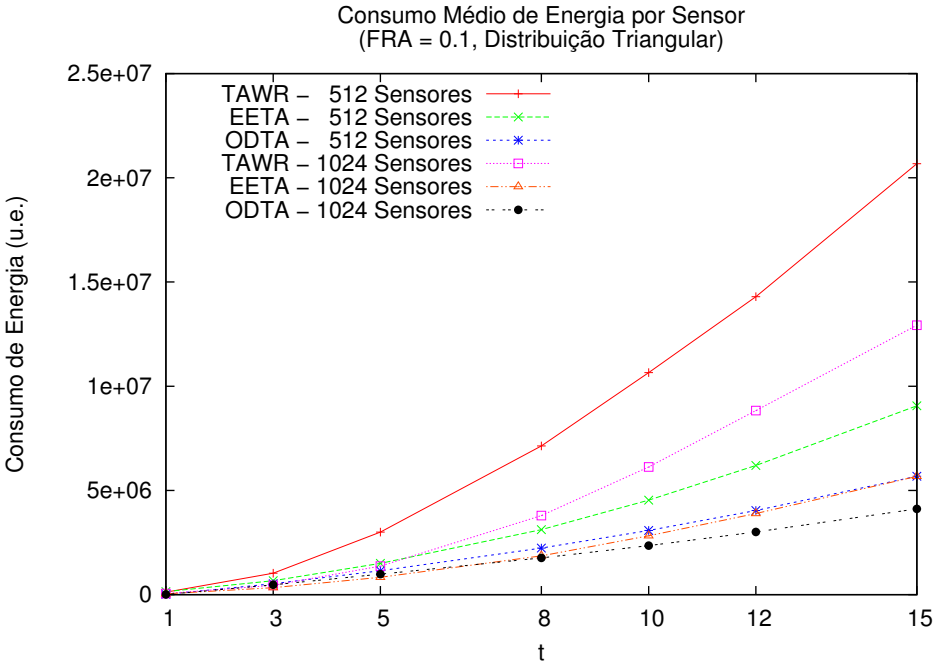


Figura 5.10: Consumo médio de energia de cada estratégia, para redes de 512 e 1024 sensores e distribuição triangular.

diferentes valores de FRA. Para $FRA = 1$ e $t = 15$, por exemplo, aproximadamente 80% dos sensores da rede são utilizados pela estratégia TAWR, gerando um gasto energético total muito maior.

5.2.3.3 Custos Energéticos de EETA e ODTA para Diferentes Valores de FRA

A Figura 5.15 mostra a comparação entre os custos totais obtidos pelas estratégias EETA e ODTA para diferentes valores de FRA. Com o crescimento do valor de FRA, a estratégia EETA obtém um aumento na eficiência energética em comparação com a estratégia ODTA. Inclusive, para $FRA = 1$, EETA apresenta valores menores do que ODTA.

Apesar de utilizar um número maior de sensores, a estratégia EETA permite que, para $FRA = 1$, custos totais menores sejam obtidos. Como visto no Capítulo 4, na estratégia ODTA cada sensor testa t sensores e é testado por t outros sensores, desta forma, ignorando o sentido dos testes, o grafo de testes D formado é um grafo completo. Por isso,

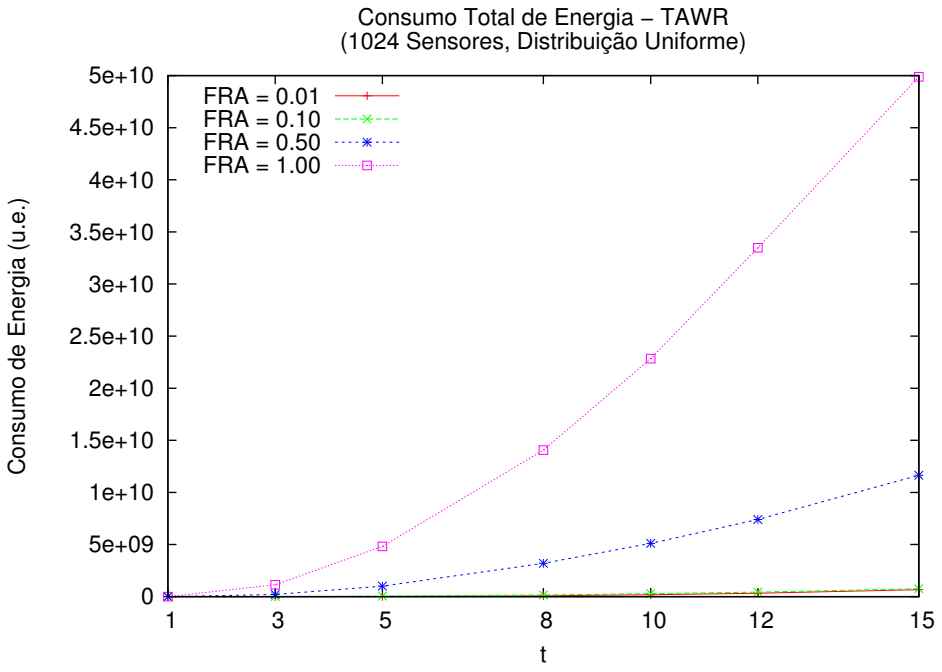


Figura 5.11: Consumo total de energia da estratégia TAWR para diferentes valores de FRA.

com sensores de T distantes entre si, testes com um alto custo energético são realizados, visto que mesmo sensores distantes terão que realizar testes entre si. A estratégia EETA permite que testes longos e de alto custo sejam, de certa forma, "quebrados" em testes menores, de menor custo. Assim, um conjunto com mais testes de baixo consumo gera custos totais menores do que um conjunto de poucos testes de alto consumo. Isso explica os valores menores de EETA para $FRA = 1$ comparados com ODTA.

O custo energético máximo ($C_{max}(D)$) obtido pelas estratégias EETA e ODTA para diferentes valores de FRA é exibido na Figura 5.16. É possível perceber que, a partir de valores de FRA maiores do que $0,1$ a estratégia EETA começa a exibir uma distribuição mais homogênea de consumo energético entre os sensores, pois apresenta valores reduzidos de consumo máximo por sensor. Isso se deve ao fato da abordagem EETA gerar um conjunto com mais testes, porém de menor custo. A abordagem ODTA, por outro lado, possui sensores com maiores consumos de energia. Tais sensores, em grande parte dos casos, pertencem a T e, quando FRA é suficientemente alto, estão mais afastados dos demais sensores.

As Figuras 5.17 e 5.18 mostram a aplicação das estratégias EETA e ODTA para um

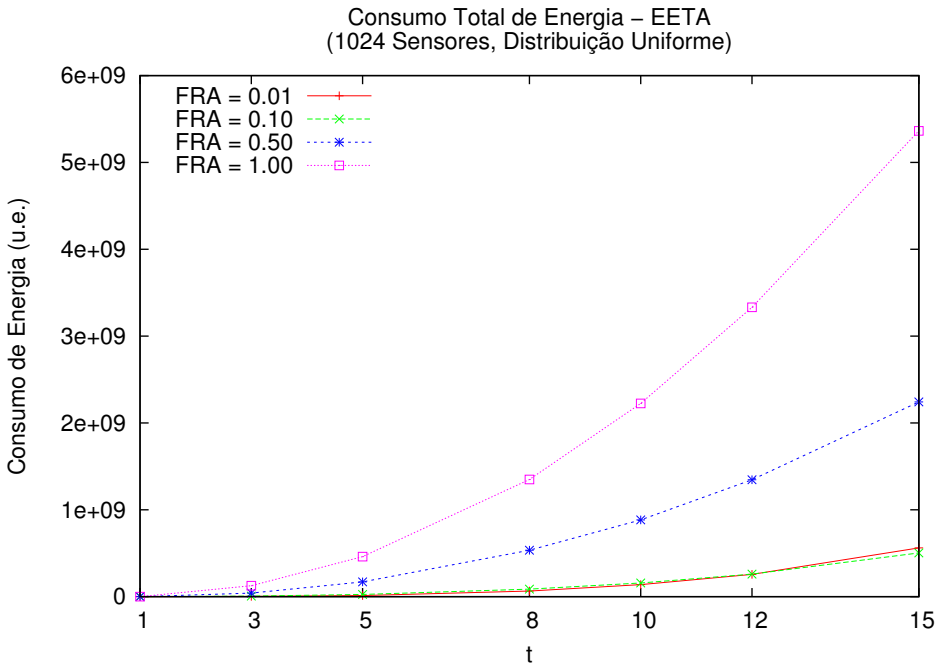


Figura 5.12: Consumo total de energia da estratégia EETA para diferentes valores de FRA.

mesmo experimento, com FRA=1 e $t = 3$. Nas Figuras, as arestas representam testes executados, e os sensores marcados com um "X" representam os sensores de T. Círculos escuros representam sensores de V_D. Através das Figuras é possível notar que a estratégia EETA utiliza mais sensores e testes, porém gera testes de menor custo. A estratégia ODTA, por sua vez, em seu grafo completo, acaba gerando testes longos e de alto consumo energético para testar, entre outros, os sensores de T que estão distantes entre si. Tais testes proporcionam um crescimento no consumo total utilizado pela estratégia.

5.3 Discussão

Os resultados dos experimentos apresentados neste Capítulo permitem algumas conclusões no que diz respeito à escolha das estratégias para determinadas aplicações. São avaliados aqui, baseado nos resultados obtidos, pontos positivos e negativos de cada estratégia.

A estratégia TAWR apresenta um consumo de energia superior ao das demais estratégias em todos os casos apresentados. O número de sensores utilizados pela estratégia também torna o diagnóstico pesado em termos de comunicação entre os sensores e de

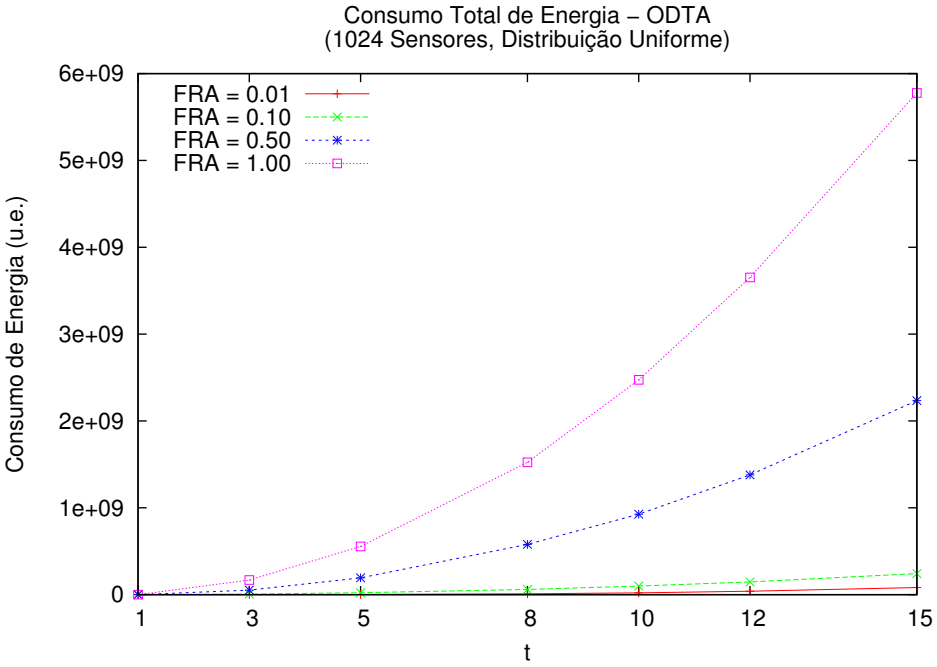


Figura 5.13: Consumo total de energia da estratégia ODTA para diferentes valores de FRA.

escalabilidade. Porém para valores baixos de t a estratégia apresenta um comportamento de menor custo energético, que se mostra não muito superior às demais abordagens. A estratégia também possui propriedade distribuída, onde os sensores realizam a formação do grafo de testes, que pode ser desejável em certas aplicações, como, por exemplo, em situações em que o acesso ao *sink* é uma operação de difícil execução pelos sensores, ou de alto custo, e que deve ser evitada.

Por sua vez, as estratégias EETA e ODTA apresentam custos energéticos menores. A estratégia ODTA mostra ser a opção de menor custo para a maioria dos casos. Além disso, utiliza um menor número de sensores, o que pode aumentar o tempo de vida útil da rede. Já a estratégia EETA, apesar de utilizar um número maior de sensores, permite, em alguns casos, a utilização de testes de menor custo energético e uma melhor distribuição de gasto de energia entre os sensores.

Para aplicações com comportamento aleatório de alarmes, ou seja, aplicações em que um mesmo fenômeno pode ser detectado por sensores distantes entre si (alto FRA), o uso da estratégia TAWR não se mostra uma boa opção com relação ao gasto energético e ao número de mensagens trocadas. A partir de casos com FRA maiores que $0,5$, a

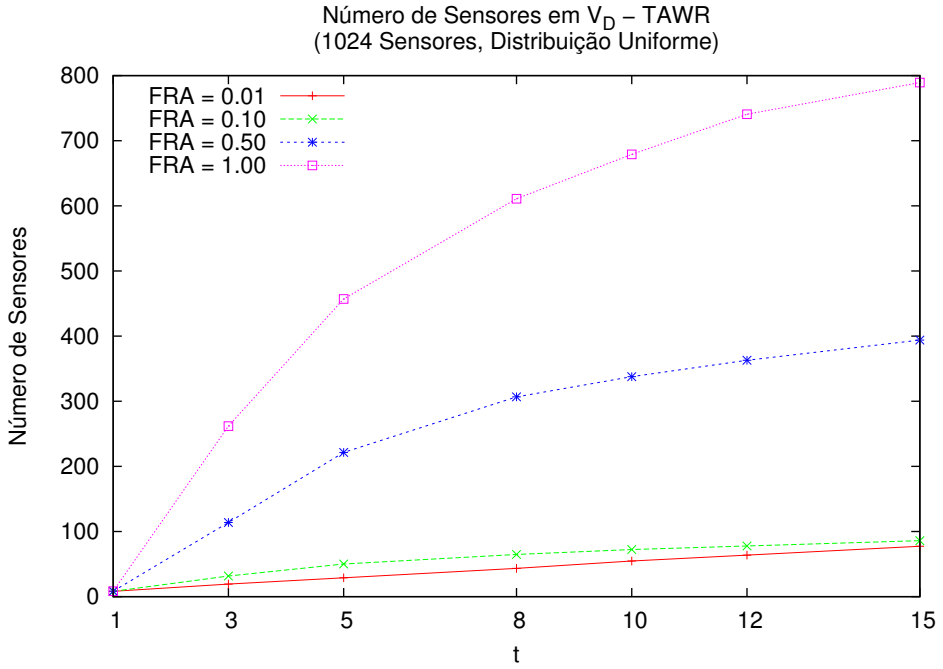


Figura 5.14: Número de sensores utilizados pela estratégia TAWR para diferentes valores de FRA.

estratégia EETA se mostra uma opção melhor em termos de consumo total de energia do que a estratégia ODTA.

Assim, a estratégia ODTA se mostra a mais eficiente em termos de energia e de escalabilidade para a maioria dos casos, devido ao baixo número de sensores e de testes utilizados. Apesar da estratégia EETA apresentar um consumo relativamente menor em alguns casos, a estratégia ODTA permite um melhor desempenho em aplicações reais, pois, de uma forma geral, em tais aplicações, a ocorrência de um comportamento como o relatado com $AFR = 1$ é raro. Fenômenos monitorados por sensores iniciam-se de forma localizada e, desta forma, serão detectados por sensores sem-falha próximos entre si. A Figura 5.19 mostra o grafo de testes final gerado por cada estratégia para um mesmo caso, onde $t = 3$. Na Figura, percebe-se que a estratégia ODTA apresenta uma redução significativa do número sensores utilizados, representados por círculos pretos, e, consequentemente, de testes, representados por arestas.

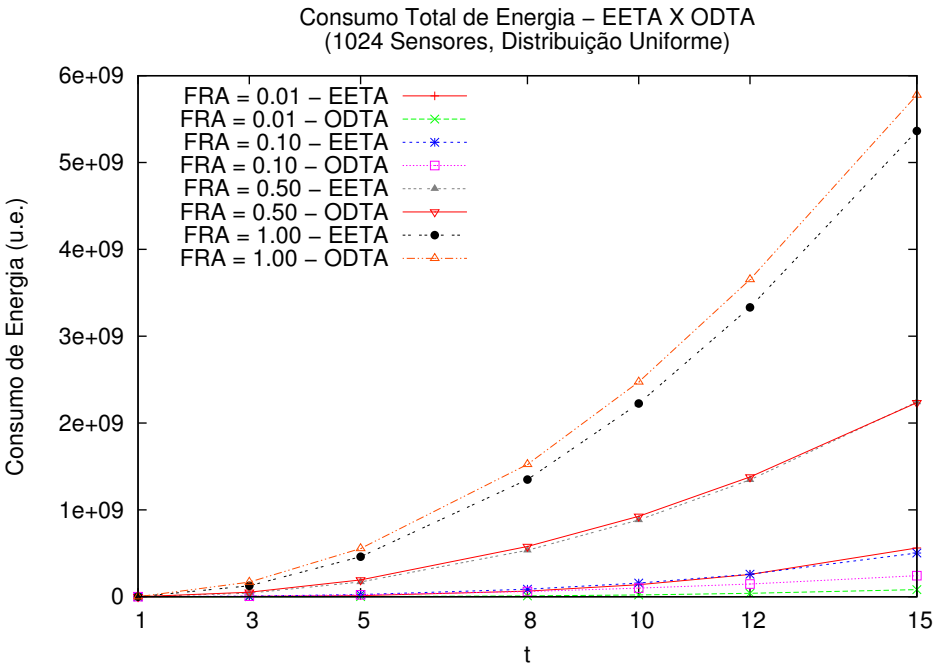


Figura 5.15: Consumo total de energia das estratégias EETA e ODTA para diferentes valores de FRA.

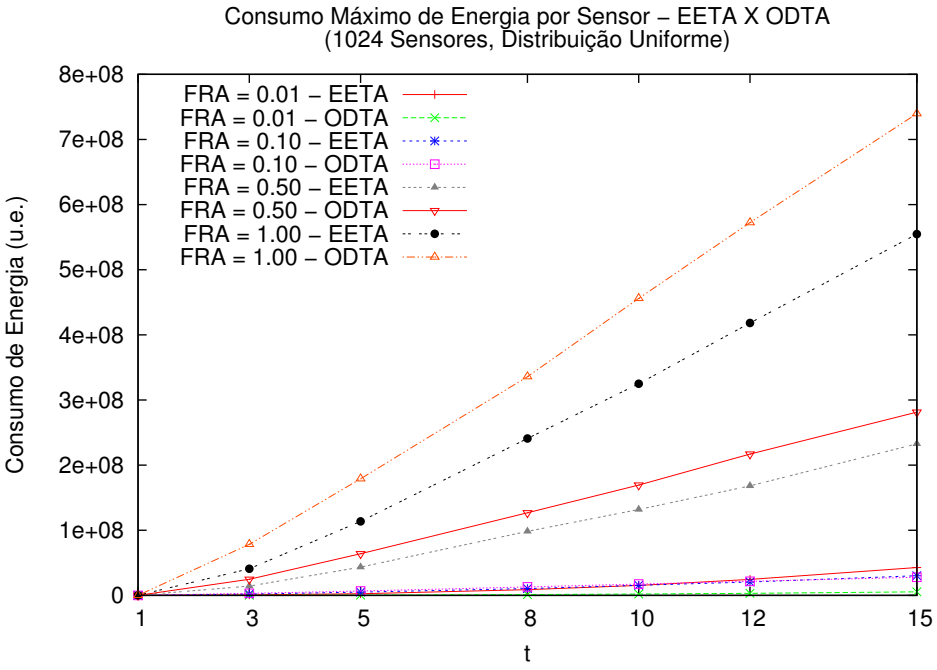


Figura 5.16: Consumo máximo de energia por sensor das estratégias EETA e ODTA para diferentes valores de FRA.

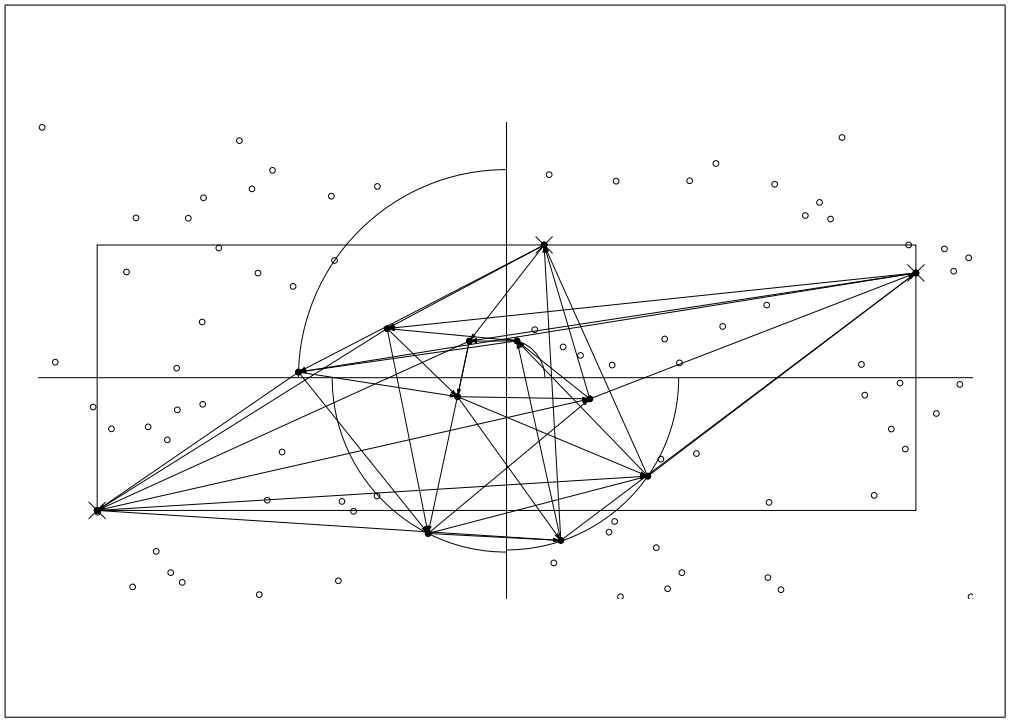


Figura 5.17: Exemplo de aplicação da estratégia EETA para FRA=1 e $t = 3$.

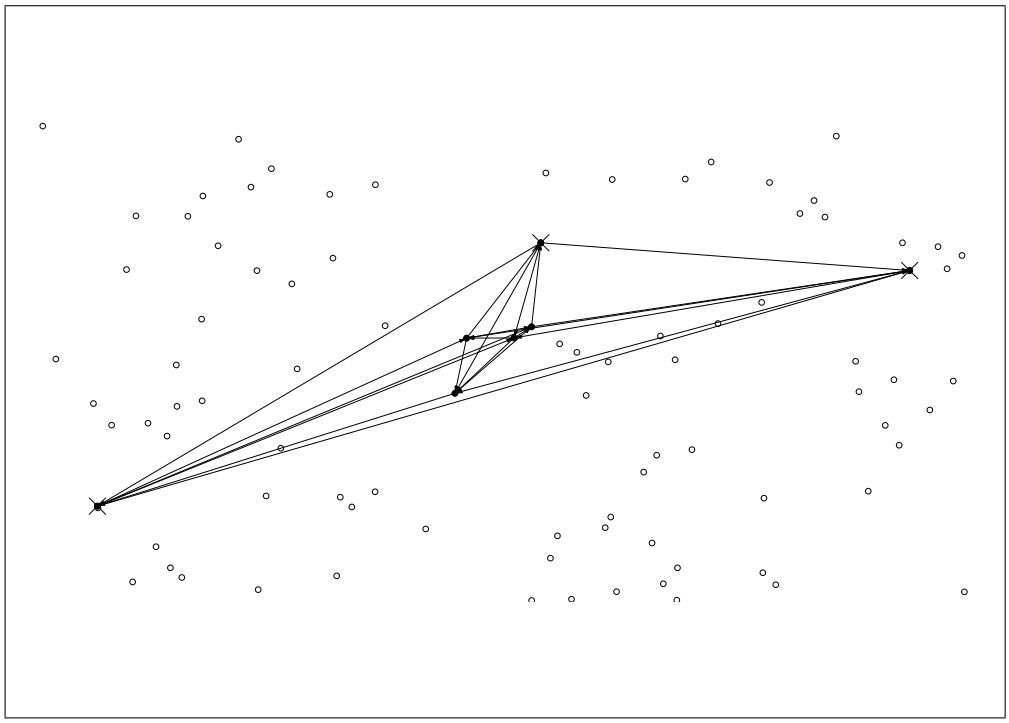
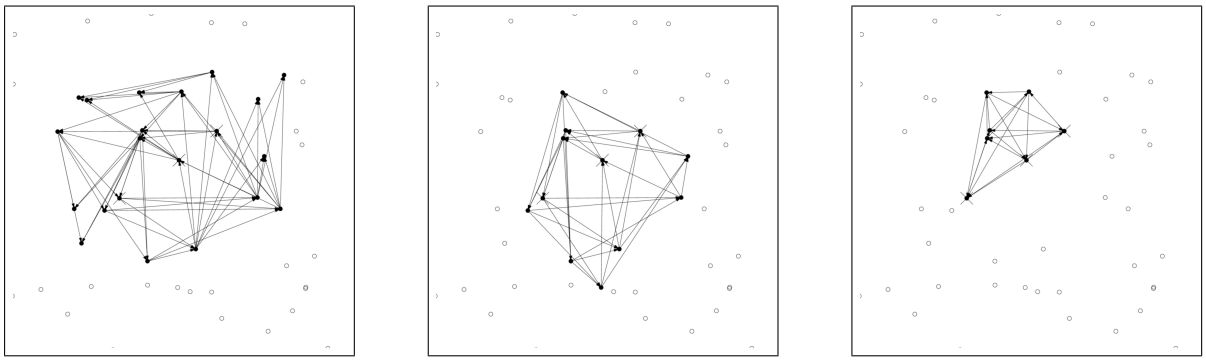


Figura 5.18: Exemplo de aplicação da estratégia ODTA para FRA=1 e $t = 3$.



(a) TAWR (b) EETA (c) ODTA

Figura 5.19: Comparação do grafo de testes gerado por cada estratégia para um mesmo caso, com $t = 3$.

CAPÍTULO 6

CONCLUSÃO

Neste trabalho, foram descritas e comparadas abordagens de testes para um algoritmo de diagnóstico em nível de sistema. As abordagens apresentadas visam o problema da detecção de alarmes falsos (falsos positivos) em uma rede de sensores sem fio onde os sensores monitoram o ambiente com o objetivo de gerar alarmes sobre a ocorrência de determinados eventos. Foram descritas três abordagens: TAWR, EETA e ODTA. Em cada uma delas, assume-se que t sensores geraram alarmes. A abordagem TAWR é baseada em um algoritmo distribuído onde sensores escolhidos para participar do diagnóstico solicitam testes entre si seguindo uma ordem pré-estabelecida e, desta forma, geram o assinalamento de testes. A abordagem EETA, por sua vez, é baseada na escolha iterativa de um conjunto de $4t$ sensores que produzem um assinalamento de testes, definido pelo *sink*, de menor custo energético total. E, por fim, na abordagem ODTA, o *sink* localiza $t + 1$ sensores próximos da região de onde o alarme foi gerado possibilitando, assim, a formação de um assinalamento de testes com $2t + 1$ sensores. As abordagens tiveram seus consumos energéticos comparados através de simulações. A abordagem TAWR apresentou um consumo energético total superior às demais estratégias em todos os experimentos. As abordagens EETA e ODTA apresentaram custos energéticos totais menores devido ao número reduzido de sensores utilizados no processo. Embora ambas estratégias EETA e ODTA tenham apresentado custos energéticos totais menores que TAWR, a estratégia ODTA apresentou os menores consumos de energia para a maioria dos casos simulados. Simulações de redes geradas com distribuição triangular apresentam uma melhora ainda mais acentuada da estratégia ODTA para ambientes de maior concentração de sensores.

Como trabalhos futuros, pretende-se avaliar outras distribuições de probabilidade para a deposição dos sensores na rede, bem como a escalabilidade das soluções apresentadas. A comparação com outras abordagens de diagnóstico também está prevista.

BIBLIOGRAFIA

[1] P. Baronti, P. Pillai, V. W. C. Chook, S. Chessa, A. Gotta, e Y. F. Hu, "Wireless Sensor Networks: A Survey on the State of the Art and the 802.15.4 and ZigBee Standards," *Computer Communications*, volume 30, n^o 2, página 1655–1695, 2007.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, e E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, volume 38, n^o 4, página 393–422, 2002.

[3] M. Barborak, A. Dahbura, e M. Malek, "The Consensus Problem in Fault-Tolerant Computing," *ACM Computer Surveys.*, volume 25, página 171–220, 1993.

[4] M. Malek, "A Comparison Connection Assignment for Diagnosis of Multiprocessor Systems," in *Proceedings of the 7th Symposium on Computer Architecture*, ISCA '80, página 31–36, 1980.

[5] K.-Y. Chwa e S. L. Hakimi, "Schemes for Fault-Tolerant Computing: A Comparison of Modularly Redundant and t-Diagnosable Systems," *Information and Control*, volume 49, n^o 3, página 212–238, 1981.

[6] J. Maeng e M. Malek, "A Comparison Connection Assignment for Self-Diagnosis of Multiprocessor Systems," in *Proceedings of the 11th International Symposium of Fault-Tolerant Computing*, FTCS '81, página 173–175, 1981.

[7] A. Avizienis, J.-C. Laprie, B. Randell, e C. Landwehr, "Basic Concepts and Taxonomy of Dependable and Secure Computing," *IEEE Transactions on Dependable and Secure Computing*, volume 1, n^o 1, página 11–33, 2004.

[8] F. P. Preparata, G. Metze, e R. T. Chien, "On the Connection Assignment Problem of Diagnosable Systems," *IEEE Transactions on Computers Electronic*, volume 16, n^o 12, página 848–854, 1967.

[9] P. Santi e S. Chessa, "Comparison-Based System Level Fault Diagnosis in Ad Hoc Networks," in *Proceedings of the 20th IEEE Symposium on Reliable Distributed Systems*, SRDS '01, página 257–266, 2001.

[10] A. Bharathidasan e V. A. S. Ponduru, "Sensor Networks: An Overview," *IEEE Potentials*, volume 22, n^o 2, página 20–23, 2003.

[11] I. Chlamtac, M. Conti, e J. J. N. Liu, "Mobile Ad Hoc Networking: Imperatives and Challenges," *Ad Hoc Networks*, volume 1, n^o 1, página 13–64, 2003.

[12] S. L. Hakimi e A. T. Amin, "Characterization of Connection Assignment of Diagnosable Systems," *IEEE Transactions on Computers*, volume 23, n^o 1, página 86–88, 1974.

[13] S. L. Hakimi e K. Nakajima, "On Adaptive System Diagnosis," *IEEE Transactions on Computers*, volume 33, n^o 3, página 234–240, 1984.

[14] S. H. Hosseini, J. G. Kuhl, e S. M. Reddy, "A Diagnosis Algorithm for Distributed Computing Systems with Dynamic Failure and Repair," *IEEE Transactions on Computers*, volume 33, n^o 3, página 223–233, 1984.

[15] R. P. Bianchini Jr. e R. W. Buskens, "An Adaptive Distributed System-Level Diagnosis Algorithm and its Implementation," in *Proceedings of the 21th International Symposium of Fault-Tolerant Computing*, FTCS '91, página 222–229, IEEE, 1991.

[16] E. P. Duarte Jr. e T. Nanya, "A Hierarchical Adaptive Distributed System-Level Diagnosis Algorithm," *IEEE Transactions on Computers*, volume 47, n^o 1, página 34–45, 2002.

[17] E. P. Duarte Jr., A. Brawerman, e L. C. P. Albini, "A Diagnosis Algorithm Based on Clusters with Detours," in *6th Latin American Network Operations and Management Symposium*, LANOMS '09, página 1–6, 2009.

[18] E. P. Duarte Jr., A. Brawerman, e L. C. P. Albini, "An Algorithm for Distributed Hierarchical Diagnosis of Dynamic Fault and Repair Events," in *Proceedings of*

the 7th International Conference on Parallel and Distributed Systems, ICPADS '00, página 299–306, IEEE Computer Society, 2000.

[19] A. Subbiah e D. M. Blough, "Distributed Diagnosis in Dynamic Fault Environments," *IEEE Transactions on Parallel and Distributed Systems*, volume 15, nº 5, página 453–467, 2004.

[20] B. Karp e H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," in *Proceedings of the 6th International Conference on Mobile Computing and Networking*, MobiCom '00, página 243–254, ACM, 2000.

[21] A. T. Dahbura e G. M. Masson, "An $0(n^{2.5})$ Fault Identification Algorithm for Diagnosable Systems," *IEEE Transactions on Computers*, volume 100, nº 6, página 486–492, 1984.

[22] A. Caruso, S. Chessa, P. Maestrini, e P. Santi, "Diagnosability of Regular Systems," *Journal of Algorithms*, volume 45, nº 2, página 126–143, 2002.

[23] A. Milenkovic, C. Otto, e E. Jovanov, "Wireless Sensor Networks for Personal Health Monitoring: Issues and an Implementation," *Computer Communications*, volume 29, nº 13-14, página 2521–2533, 2006.

[24] ZigBee Alliance, "ZigBee Wireless Sensor Applications for Health, Wellness and Fitness," tech. rep., 2009.

[25] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, e J. Zhao, "Habitat monitoring: Application driver for wireless communications technology," *ACM SIGCOMM Computer Communication Review*, volume 31, nº 2, página 20–41, 2001.

[26] N. Bulusu, J. Heidemann, e D. Estrin, "GPS-Less Low-Cost Outdoor Localization for Very Small Devices," *IEEE Personal Communications*, volume 7, nº 5, página 28–34, 2000.

[27] R. Shah e J. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks," in *Wireless Communications and Networking Conference*, vol. 1 of *InfOCOM '06*, página 350–355, 2002.

[28] J. Kulik, W. R. Heinzelman, e H. Balakrishnan, "Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks," *Wireless Networks*, volume 8, página 169–185, 1999.

[29] D. Braginsky e D. Estrin, "Rumor Routing Algorithm for Sensor Networks," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, WSNA '02, página 22–31, 2002.

[30] C. Intanagonwiwat, R. Govindan, e D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *Proceedings of the 6th International Conference on Mobile Computing and Networking*, MobiCom '00, página 56–67, 2000.

[31] D. Ganesan, R. Govindan, S. Shenker, e D. Estrin, "Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, volume 5, nº 4, página 11–25, 2001.

[32] ZigBee Alliance, "ZigBee Specification v 1.0," tech. rep., 2004.

[33] IEEE 802.15.4 Working Group, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)," Technical Standard IEEE Std 802.15.4-2003, IEEE, 2003.

[34] S. Tomic, "Network-Growing Scenarios in IEEE 802.15.4 Wireless Sensor Networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications*, InfOCOM '06, 2006.

[35] J. Hill, M. Horton, R. Kling, e L. Krishnamurthy, "The Platforms Enabling Wireless Sensor Networks," *Communications of the ACM*, volume 47, nº 6, página 41–46, 2004.

[36] F. Barsi, F. Grandoni, e P. Maestrini, "A Theory of Diagnosability of Digital Systems," *IEEE Transactions on Computers*, volume 100, n^o 25, página 585–593, 1976.

[37] A. K. Somani, V. K. Agarwal, e D. Avis, "A Generalized Theory for System Level Diagnosis," *IEEE Transactions on Computers*, volume 36, n^o 5, página 538–546, 1987.

[38] J. G. Kuhl e S. M. Reddy, "Distributed Fault-Tolerance for Large Multiprocessor Systems," in *Proceedings of the 7th Symposium on Computer Architecture*, ISCA '80, página 23–30, ACM, 1980.

[39] R. P. Bianchini Jr. e R. W. Buskens, "Implementation of On-Line Distributed System-Level Diagnosis Theory," *IEEE Transactions on Computers*, volume 41, n^o 5, página 616–626, 1992.

[40] A. Bagchi, S. L. Hakimi, e R. Bellcore, "An Optimal Algorithm for Distributed Dystem Level Diagnosis," in *21st International Symposium Fault-Tolerant Computing*, FTCS '21, página 214–221, 1991.

[41] M. Stahl, R. W. Buskens, e R. P. Bianchini Jr., "Simulation f the Adapt On-Line Diagnosis Algorithm for General Topology Networks," in *Proceedings of the IEEE 11th Symposium on Reliable Distributed Systems*, página 180–187, 1992.

[42] S. Rangarajan, A. T. Dahbura, e E. Ziegler, "A Distributed System-Level Diagnosis Algorithm for Arbitrary Network Topologies," *IEEE Transactions on Computers*, volume 44, n^o 2, página 312–334, 1995.

[43] E. P. Duarte Jr., T. Nanya, G. Mansfield, e S. Noguchi, "Non-Broadcast Betwork Fault-Monitoring Based on System-Level Diagnosis," in *Proceedings of the 5th IFIP/IEEE International Symposium on Integrated Network Management*, IM '97, página 597–609, 1997.

[44] E. P. Duarte Jr., "Um Algoritmo para Diagnóstico de Redes de Topologia Arbitrária," in *Proceedings of the 1st SBC Workshop on Test and Fault Tolerance*, SBCWTF '98, página 50–55, 1998.

[45] E. P. Duarte Jr. e A. Weber, "A Distributed Network Connectivity Algorithm," in *Proceedings of the 6th International Symposium on Autonomous Decentralized Systems*, ISADS '03, páginas 285, IEEE Computer Society, 2003.

[46] A. Weber, A. W. Santos, E. P. Duarte Jr., e K. V. O. Fonseca, "Simulação de um Algoritmo de Diagnóstico Distribuído para Redes Particionáveis de Topologia Arbitrária," in *9º Workshop de Testes e Tolerância a Falhas do 26o Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, WTF '08 - SBRC '08, página 113–126, 2008.

[47] E. P. Duarte Jr., R. P. Ziwich, e L. C. P. Albini, "A Survey of Comparison-Based System-Level Diagnosis," *ACM Computing Surveys*, volume V, nº N, 20YY. Accepted for publication.

[48] A. Sengupta e A. T. Dahbura, "On Self-Diagnosable Multiprocessor Systems: Diagnosis by the Comparison Approach," *IEEE Transactions on Computers*, volume 41, nº 11, página 1386–1396, 1992.

[49] A. T. Dahbura, K. K. Sabnani, e L. L. King, "The Comparison Approach to Multiprocessor Fault Diagnosis," *IEEE Transactions on Computers*, volume 36, nº 3, página 373–378, 1987.

[50] D. M. Blough e H. W. Brown, "The Broadcast Comparison Model for On-Line Fault Diagnosis in Multicomputer Systems: Theory and Implementation," *IEEE Transactions on Computers*, volume 48, nº 5, página 470–493, 1999.

[51] L. C. P. Albini e E. P. Duarte Jr., "Generalized Distributed Comparison-Based System-Level Diagnosis," in *2nd IEEE Latin American Test Workshop*, página 285–290, 2001.

[52] R. Pereira Ziwich, E. P. Duarte Jr., e L. C. P. Albini, "Distributed Integrity Checking for Systems with Replicated Data," in *Proceedings of the 11th International Conference on Parallel and Distributed Systems*, ICPADS '05, página 363–369, IEEE Computer Society, 2005.

[53] X. Yang e Y. Yan Tang, "Efficient Fault Identification of Diagnosable Systems under the Comparison Model," *IEEE Transactions on Computers*, volume 56, nº 12, página 1612–1618, 2007.

[54] M. Elhadef, A. Boukerche, e H. Elkadiki, "Self-Diagnosing Wireless Mesh and Ad-Hoc Networks using an Adaptable Comparison-Based Approach," in *Proceedings of the 2nd International Conference on Availability, Reliability and Security*, ARES '07, página 983–990, IEEE Computer Society, 2007.

[55] F. S. Martins, M. Maia, R. M. C. de Andrade, A. L. Santos, e J. N. Souza, "Detecting Malicious Manipulation in Grid Environments," in *18TH International Symposium on Computer Architecture and High Performance Computing*, SBAC-PAD '06, página 28–35, IEEE, 2006.

[56] X. Liu, X. Yang, e M. Xiang, "One-Step t-Fault Diagnosis for Hypermesh Optical Interconnection Multiprocessor Systems," *Journal of Systems and Software*, volume 82, nº 9, página 1491–1496, 2009.

[57] Q. Zhou, S. Liu, e Q. Zhu, "Local Diagnosability of Generic Star-Pyramid Graph," *Information Processing Letters*, volume 109, nº 13, página 695–699, 2009.

[58] J. Fan, J. Yang, G. Zhou, L. Zhao, e W. Zhang, "Diagnosable Evaluation of DCC Linear Congruential Graphs Under the PMC Diagnostic Model," *Information Sciences*, volume 179, nº 11, página 1785–1791, 2009.

[59] J. Opatrny, D. Sotteau, N. Srinivasan, e K. Thulasiraman, "DCC Linear Congruential Graphs: A New Class of Interconnection Networks," *IEEE Transactions on Computers*, volume 45, nº 2, página 156–164, 1996.

[60] M. Elhadef, "A Perceptron Neural Network for Asymmetric Comparison-Based System-Level Fault Diagnosis," in *International Conference on Availability, Reliability and Security*, ARES'09, página 265–272, IEEE, 2009.

[61] F. Rosenblatt, "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain," *Psychological review*, volume 65, nº 6, página 386–408, 1958.

[62] T. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford, UK: Oxford University Press, 1996.

[63] H. Yang, M. Elhadef, A. Nayak, e X. Yang, "An Evolutionary Approach to System-Level Fault Diagnosis," in *Proceedings of the 11th Conference on Evolutionary Computation*, CEC'09, página 1406–1413, 2009.

[64] F. S. Martins, R. M. Andrade, A. L. Santos, B. Schulze, e J. N. Souza, "Detecting Misbehaving Units on Computational Grids," *Concurrency and Computation: Practice and Experience*, volume 22, nº 3, página 329–342, 2009.

[65] S. Chessa e P. Santi, "Crash Faults Identification in Wireless Sensor Networks," *Computer Communications*, volume 25, nº 14, página 1273–1282, 2002.

[66] J.-Y. Choi, S.-J. Yim, Y. J. Huh, e Y. H. Choi, "A Distributed Adaptive Scheme for Detecting Faults in Wireless Sensor Networks," *WSEAS Transactions on Communications*, volume 8, nº 2, página 269–278, 2009.

[67] Z. Taghikhaki e M. Sharifi, "A Trust-Based Distributed Data Fault Detection Algorithm for Wireless Sensor Networks," in *11th International Conference on Computer and Information Technology*, ICCIT 2008, página 1–6, 2008.

[68] A. Sharma, L. Golubchik, e R. Govindan, "On the Prevalence of Sensor Faults in Real-World Deployments," in *4th IEEE Communications Society Conference on Sen-*

sor, Mesh and Ad Hoc Communications and Networks, SECON '07, página 213–222, IEEE, 2007.

[69] G. Venkataraman, S. Emmanuel, e S. Thambipillai, "Energy-Efficient Cluster-Based Scheme for Failure Management in Sensor Networks," *IET Communications*, volume 2, nº 4, página 528–537, 2008.

[70] A. Weber, A. R. Kutzke, e S. Chessa, "Diagnosability Evaluation for a System-Level Diagnosis Algorithm for Wireless Sensor Networks," in *IEEE Symposium on Computers and Communications*, ISCC '10, página 241–244, 2010.

[71] A. Weber, A. R. Kutzke, e S. Chessa, "Energy-Aware Test Connection Assignment for the Diagnosis of a Wireless Sensor Network," in *Proeedings of 5th Latin American Symposium on Dependable Computing*, LADC '11, 2011. A ser publicado.

[72] R. Schlichting e F. Schneider, "Fail-Stop Processors: An Approach to Designing Fault-Tolerant Computing Systems," *ACM Transactions on Computer Systems (TOCS)*, volume 1, nº 3, página 222–238, 1983.

[73] F. Cristian, H. Aghili, R. Strong, e D. Dolev, "Atomic Broadcast: From Simple Message Diffusion to Byzantine Agreement," *Information and Computation*, volume 118, nº 1, páginas 158, 1985.

[74] L. A. Laranjeira, M. Malek, e R. Jenevein, "On Tolerating Faults in Naturally Redundant Algorithms," in *Proceedings of the 10th Symposium on Reliable Distributed Systems*, página 118–127, IEEE, 1991.

[75] L. Lamport, R. Shostak, e M. Pease, "The Byzantine Generals Problem," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, volume 4, nº 3, página 382–401, 1982.

[76] M. Elhadef, A. Boukerche, e H. Elkadiki, "Performance Analysis of a Distributed Comparison-Based Self-Diagnosis Protocol for Wireless ad-hoc Networks," in *Procee-*

dings of the 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems, página 165–172, ACM, 2006.

[77] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, e R. O'dea, "Relative Location Estimation in Wireless Sensor Networks," *IEEE Transactions on Signal Processing*, volume 51, n^o 8, página 2137–2148, 2003.

[78] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall, 2nd ed., 2001.

[79] M. D. Penrose, "On K-Connectivity for a Geometric Random Graph," *Random structures and Algorithms*, volume 15, n^o 2, página 145–164, 1999.

[80] M. Evans, N. Hastings, e B. Peacock, *Statistical Distributions*, ch. Triangular distribution, página 187–188. 3 ed.